Multi-guide Particle Swarm Optimization for Many-objective Optimization Problems

by

Cian Steenkamp

Thesis presented in partial fulfilment of the requirements for the degree of Master of Science (Computer Science) in the Faculty of Computer Science at Stellenbosch University

Supervisor: Prof. A. P. Engelbrecht

March 2021

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: 2021/01/26

Abstract

Multi-guide Particle Swarm Optimization for Many-objective Optimization Problems

C. Steenkamp

Department of Computer Science,
University of Stellenbosch,
Private Bag X1, Matieland 7602, South Africa.

Thesis: MSc (CS)

January 2021

The scalability of the multi-guide particle swarm optimization (MGPSO) algorithm, with respect to the number of objectives for a problem, is investigated. Two MGPSO algorithm adaptations are proposed; that is, the partial-dominance multi-guide particle swarm optimization (PMGPSO) algorithm and the knee-point driven multi-guide particle swarm optimization (KnMGPSO) algorithm. As a sub-objective, the effect of different archive balance coefficient update strategies for the MGPSO, the PMGPSO, and the KnMGPSO algorithms are investigated. The proposed algorithms attempt to address the scalability limitations associated with a certain component of the MGPSO algorithm. This study does not consider scalability with respect to the number of decision variables. This study assumes a static search space; that is, where the number of objectives remains fixed throughout the optimization. This study also assumes that each objective remains static throughout the search process. This study also considers only problems with boundary constraints. The results indicate that the MGPSO algorithm scaled to many-objectives competitively compared to other state-of-the-art many-objective optimization algorithms. The results were unexpected because the MGPSO algorithm uses the Pareto-dominance relation, which is known to degrade as the number of objectives increases. The proposed PMGPSO and KnMGPSO algorithms also scaled competitively, however, these algorithms were not superior to the MGPSO algorithm. The investigated dynamic archive balance coefficient update strategies did not improve the performance of the MGPSO, the PMGPSO, or the KnMGPSO algorithms.

Keywords: Multi-guide particle swarm optimization, particle swarm optimization, many-objective optimization, scalability

Acknowledgements

I would like to express my sincere gratitude to the following people and organisations, without whom this work would not have been possible:

- My supervisor, Prof. Andries Engelbrecht, for his guidance, mentorship, emotional support at times, and continued insight and inspiration.
- Thomas Fogwill and Matthew Chetty at the CSIR for all the opportunities that they have presented to me.
- Kyle Erwin who helped me extensively in terms of algorithm understanding and implementation.
- The Centre for High Performance Computing (CHPC) for allowing me to run my simulations on their infrastructure.
- Dr Erik Hofstee for teaching me the tools to be able to construct a good dissertation.
- My family and friends for emotional support.
- Isjeanne du Toit, the love of my life, for being my best friend and for motivating me.

This work has been supported by a studentship from the Council for Scientific and Industrial Research (CSIR).

Dedications

"If you go out looking for friends, you're going to find that they are scarce.
If you go out to be a friend, you'll find them everywhere."

— Zig Ziglar

This dissertation is dedicated to two of my best friends, Izak Celliers and Chris Snyman, who have stepped into glory. We miss you and can't wait to be reunited.

Contents

Declaration **i**

Abstract **ii**

Acknowledgements **iv**

Dedications **v**

Contents **vi**

List of Algorithms **ix**

List of Figures **x**

List of Tables **xi**

1 Introduction **1**
1.1 Motivation 1
1.2 Research Objectives 2
1.3 Contributions 3
1.4 Dissertation Outline 4

2 Many-objective Optimization **7**
2.1 Multi-objective Optimization 9
2.1.1 Multi-objective Problem 9
2.1.2 Weighted Aggregation Approach 10
2.1.3 Pareto-optimality Approach 11
2.2 Many-objective Optimization 14
2.2.1 Many-objective Problem 14
2.2.2 Many-objective Optimization Challenges 14
2.3 Test Problems 17
2.3.1 Deb-Thiele-Laumanns-Zitzler Test Problems 17
2.3.2 Walking Fish Group Test Problems 29
2.4 Performance Measures 48
2.4.1 Hypervolume 48

2.4.2 Inverted Generational Distance 49
2.5 Summary 49

3 Many-objective Optimization Algorithms 50
3.1 Evolutionary Algorithms 51
3.1.1 Basic Evolutionary Algorithm 51
3.1.2 Many-objective Evolutionary Algorithms 52
3.1.2.1 Knee-point driven Evolutionary Algorithm . . 53
3.1.2.2 Many-objective Evolutionary Algorithm based on Dominance and Decomposition 57
3.1.2.3 Reference-point based Many-objective Non-dominated Sorting Genetic Algorithm 59
3.2 Particle Swarm Optimization 62
3.2.1 Basic Particle Swarm Optimization 62
3.2.1.1 Position and Velocity Update Equations 63
3.2.1.2 Velocity Clamping 65
3.2.1.3 Particle Swarm Optimization with Inertia Weight Coefficient 66
3.2.1.4 Particle Swarm Optimization with Constriction Coefficient 67
3.2.1.5 Neighbourhood Topologies 67
3.2.1.6 Parameter Sensitivity and Algorithm Stability 69
3.2.2 Many-objective Particle Swarm Optimization 70
3.2.2.1 Controlling Dominance Area of Solutions Speed Constraint Multi-objective Particle Swarm Optimization 71
3.2.2.2 Multi-guide Particle Swarm Optimization . . . 73
3.3 Summary 80

4 Partial-dominance Multi-guide Particle Swarm Optimization 81
4.1 Partial-dominance Approach 81
4.2 Partial-dominance Multi-guide Particle Swarm Optimization . . 82
4.3 Empirical Process 83
4.3.1 Benchmark Functions 83
4.3.2 Algorithms and Parameter Tuning Approach 84
4.3.3 Performance Measures 88
4.3.4 Statistical Analysis Process 89
4.4 Results and Discussion 90
4.4.1 Hypervolume Discussion 126
4.4.2 Inverted Generational Distance Discussion 128
4.4.3 General Discussion 130
4.5 Summary 131

5 Knee-point driven Multi-guide Particle Swarm Optimization 136

5.1 Knee-points Approach 136
5.2 Knee-point driven Multi-guide Particle Swarm Optimization . . 137
5.3 Empirical Process 139
5.3.1 Algorithms and Parameter Tuning Approach 139
5.4 Results and Discussion 140
5.4.1 Hypervolume Discussion 177
5.4.2 Inverted Generational Distance Discussion 178
5.4.3 General Discussion 180
5.5 Summary 181

6 Partial-dominance versus Knee-points 186
6.1 Empirical Process 186
6.1.1 Algorithms and Parameter Tuning Approach 186
6.2 Results and Discussion 187
6.2.1 Hypervolume Discussion 208
6.2.2 Inverted Generational Distance Discussion 209
6.2.3 General Discussion 209
6.3 Summary 211

7 Conclusions 215
7.1 Summary 215
7.2 Future Research 217

Bibliography 219

Appendices 235

A Acronyms 236

B Symbols 240

C Parameter Configurations 246

D Performance Measure Values for Chapter 4 250
D.1 Hypervolume Values 250
D.2 Inverted Generational Distance Values 281

E Performance Measure Values for Chapter 5 311
E.1 Hypervolume Values 311
E.2 Inverted Generational Distance Values 342

F Performance Measure Values for Chapter 6 372
F.1 Hypervolume Values 372
F.2 Inverted Generational Distance Values 391

G Derived Publications 409

List of Algorithms

1 Evolutionary Algorithm (EA) 52
2 Knee-point driven Evolutionary Algorithm (KnEA) 56
3 Many-objective Evolutionary Algorithm based on Dominance and Decomposition (MOEA/DD) 59
4 Reference-point based many-objective Non-dominated Sorting Genetic Algorithm (NSGA-III) 61
5 Particle Swarm Optimization (PSO) 68
6 Controlling Dominance Area of Solutions Speed Constraint Multi-objective Particle Swarm Optimization (CDAS-SMPSO) 74
7 Multi-guide Particle Swarm Optimization (MGPSO) - Archive Insert 75
8 Multi-guide Particle Swarm Optimization (MGPSO) - Search . 78
9 Multi-guide Particle Swarm Optimization (MGPSO) - Initialization 79

List of Figures

2.1 Decision and objective space. 9
2.2 Illustration of Pareto-dominance. 12
2.3 Pareto-optimal front. 12
2.4 Illustration of Ideal ($\mathbf{z}^*$) and Nadir ($\mathbf{z}^{\mathrm{nad}}$) objective vectors. 13
2.5 3-objective DTLZ1 Pareto Front 22
2.6 3-objective DTLZ2 Pareto Front 23
2.7 3-objective DTLZ3 Pareto Front 24
2.8 3-objective DTLZ4 Pareto Front 25
2.9 3-objective DTLZ5 Pareto Front 26
2.10 3-objective DTLZ6 Pareto Front 27
2.11 3-objective DTLZ7 Pareto Front 28
2.12 3-objective WFG1 Pareto Front 39
2.13 3-objective WFG2 Pareto Front 40
2.14 3-objective WFG3 Pareto Front 41
2.15 3-objective WFG4 Pareto Front 42
2.16 3-objective WFG5 Pareto Front 43
2.17 3-objective WFG6 Pareto Front 44
2.18 3-objective WFG7 Pareto Front 45
2.19 3-objective WFG8 Pareto Front 46
2.20 3-objective WFG9 Pareto Front 47

3.1 Illustration for determining knee-points. 54
3.2 Geometrical Illustration of Velocity and Position Updates for a Single Two-Dimensional Particle 64
3.3 The Star and Ring Neighbourhood Topologies 69
3.4 Illustration of CDAS for a maximization MOP with two objectives. 72

5.1 An illustrative example where weighted distance may be advantageous over crowding distance. 138

List of Tables

2.1 DTLZ Benchmark Function Properties . . . 21
2.2 WFG Benchmark Function Properties . . . 38

4.1 HV Ranking for 3-objective DTLZ . . . 90
4.2 HV Ranking for 5-objective DTLZ . . . 92
4.3 HV Ranking for 8-objective DTLZ . . . 94
4.4 HV Ranking for 10-objective DTLZ . . . 96
4.5 HV Ranking for 15-objective DTLZ . . . 98
4.6 HV Ranking for 3-objective WFG . . . 99
4.7 HV Ranking for 5-objective WFG . . . 101
4.8 HV Ranking for 8-objective WFG . . . 103
4.9 HV Ranking for 10-objective WFG . . . 105
4.10 HV Ranking for 15-objective WFG . . . 107
4.11 IGD Ranking for 3-objective DTLZ . . . 108
4.12 IGD Ranking for 5-objective DTLZ . . . 110
4.13 IGD Ranking for 8-objective DTLZ . . . 112
4.14 IGD Ranking for 10-objective DTLZ . . . 114
4.15 IGD Ranking for 15-objective DTLZ . . . 116
4.16 IGD Ranking for 3-objective WFG . . . 117
4.17 IGD Ranking for 5-objective WFG . . . 119
4.18 IGD Ranking for 8-objective WFG . . . 121
4.19 IGD Ranking for 10-objective WFG . . . 123
4.20 IGD Ranking for 15-objective WFG . . . 125
4.21 HV Ranking Summary . . . 132
4.22 IGD Ranking Summary . . . 132
4.23 HV Ranking Averages . . . 133
4.24 IGD Ranking Averages . . . 134

5.1 HV Ranking for 3-objective DTLZ . . . 141
5.2 HV Ranking for 5-objective DTLZ . . . 142
5.3 HV Ranking for 8-objective DTLZ . . . 144
5.4 HV Ranking for 10-objective DTLZ . . . 146
5.5 HV Ranking for 15-objective DTLZ . . . 148
5.6 HV Ranking for 3-objective WFG . . . 150

5.7 HV Ranking for 5-objective WFG 151
5.8 HV Ranking for 8-objective WFG 153
5.9 HV Ranking for 10-objective WFG 155
5.10 HV Ranking for 15-objective WFG 157
5.11 IGD Ranking for 3-objective DTLZ 159
5.12 IGD Ranking for 5-objective DTLZ 160
5.13 IGD Ranking for 8-objective DTLZ 162
5.14 IGD Ranking for 10-objective DTLZ 164
5.15 IGD Ranking for 15-objective DTLZ 166
5.16 IGD Ranking for 3-objective WFG 168
5.17 IGD Ranking for 5-objective WFG 169
5.18 IGD Ranking for 8-objective WFG 171
5.19 IGD Ranking for 10-objective WFG 173
5.20 IGD Ranking for 15-objective WFG 175
5.21 HV Ranking Summary 181
5.22 IGD Ranking Summary 182
5.23 HV Ranking Averages 183
5.24 IGD Ranking Averages 184

6.1 HV Ranking for 3-objective DTLZ 187
6.2 HV Ranking for 5-objective DTLZ 188
6.3 HV Ranking for 8-objective DTLZ 189
6.4 HV Ranking for 10-objective DTLZ 190
6.5 HV Ranking for 15-objective DTLZ 191
6.6 HV Ranking for 3-objective WFG 193
6.7 HV Ranking for 5-objective WFG 194
6.8 HV Ranking for 8-objective WFG 195
6.9 HV Ranking for 10-objective WFG 196
6.10 HV Ranking for 15-objective WFG 197
6.11 IGD Ranking for 3-objective DTLZ 198
6.12 IGD Ranking for 5-objective DTLZ 199
6.13 IGD Ranking for 8-objective DTLZ 200
6.14 IGD Ranking for 10-objective DTLZ 201
6.15 IGD Ranking for 15-objective DTLZ 202
6.16 IGD Ranking for 3-objective WFG 203
6.17 IGD Ranking for 5-objective WFG 204
6.18 IGD Ranking for 8-objective WFG 205
6.19 IGD Ranking for 10-objective WFG 206
6.20 IGD Ranking for 15-objective WFG 207
6.21 HV Ranking Summary 210
6.22 IGD Ranking Summary 211
6.23 HV Ranking Averages 212
6.24 IGD Ranking Averages 213

C.1 Tuned Parameter Values for the KnEA 247
C.2 Tuned Parameter Values for the MOEA/DD 248
C.3 Tuned Parameter Values for the CDAS-SMPSO algorithm 249

D.1 Average, Standard Deviation, Maximum, and Minimum HV for the CDAS-SMPSO algorithm 251
D.2 Average, Standard Deviation, Maximum, and Minimum HV for the KnEA 254
D.3 Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_R$ 257
D.4 Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{RI}$ 260
D.5 Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{STD}$ 263
D.6 Average, Standard Deviation, Maximum, and Minimum HV for the MOEA/DD 266
D.7 Average, Standard Deviation, Maximum, and Minimum HV for the NSGA-III 269
D.8 Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_R$ 272
D.9 Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{RI}$ 275
D.10 Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{STD}$ 278
D.11 Average, Standard Deviation, Maximum, and Minimum IGD for the CDAS-SMPSO algorithm 281
D.12 Average, Standard Deviation, Maximum, and Minimum IGD for the KnEA 284
D.13 Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_R$ 287
D.14 Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{RI}$ 290
D.15 Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{STD}$ 293
D.16 Average, Standard Deviation, Maximum, and Minimum IGD for the MOEA/DD 296
D.17 Average, Standard Deviation, Maximum, and Minimum IGD for the NSGA-III 299
D.18 Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_R$ 302
D.19 Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{RI}$ 305
D.20 Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{STD}$ 308

E.1 Average, Standard Deviation, Maximum, and Minimum HV for the CDAS-SMPSO algorithm . . . 312
E.2 Average, Standard Deviation, Maximum, and Minimum HV for the KnEA . . . 315
E.3 Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_R$ . . . 318
E.4 Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{RI}$ . . . 321
E.5 Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{STD}$ . . . 324
E.6 Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_R$ . . . 327
E.7 Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{RI}$ . . . 330
E.8 Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{STD}$ . . . 333
E.9 Average, Standard Deviation, Maximum, and Minimum HV for the MOEA/DD . . . 336
E.10 Average, Standard Deviation, Maximum, and Minimum HV for the NSGA-III . . . 339
E.11 Average, Standard Deviation, Maximum, and Minimum IGD for the CDAS-SMPSO algorithm . . . 342
E.12 Average, Standard Deviation, Maximum, and Minimum IGD for the KnEA . . . 345
E.13 Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_R$ . . . 348
E.14 Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{RI}$ . . . 351
E.15 Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{STD}$ . . . 354
E.16 Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_R$ . . . 357
E.17 Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{RI}$ . . . 360
E.18 Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{STD}$ . . . 363
E.19 Average, Standard Deviation, Maximum, and Minimum IGD for the MOEA/DD . . . 366
E.20 Average, Standard Deviation, Maximum, and Minimum IGD for the NSGA-III . . . 369

F.1 Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_R$ . . . 373

F.2 Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{RI}$ . . . 376
F.3 Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{STD}$ . . . 379
F.4 Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{R}$ . . . 382
F.5 Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{RI}$ . . . 385
F.6 Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{STD}$ . . . 388
F.7 Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{R}$ . . . 391
F.8 Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{RI}$ . . . 394
F.9 Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{STD}$ . . . 397
F.10 Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{R}$ . . . 400
F.11 Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{RI}$ . . . 403
F.12 Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{STD}$ . . . 406

Chapter 1

Introduction

"If everything seems under control, you're not going fast enough."
— Mario Gabriele Andretti, *former F1 champion.*

1.1 Motivation

Numerous real-life optimization problems consist of multiple conflicting objectives that need to be optimized. These problems are called multi-objective optimization problems (MOPs). MOPs are frequently encountered in domains such as engineering [175], business [16], mathematics [89], and physics [92]. Consider the Coronavirus disease (COVID-19) as an example of an arduous real-world MOP with (at least) three, clearly conflicting, objectives; that is, to simultaneously maximize economic (or social) activity, minimize the number of lives lost, and minimize the spread of the virus. The solution to a MOP is not a single solution, but rather a collection of optimal trade-offs. Multi-objective optimization (MOO) algorithms are used to find these well-balanced solutions.

Many-objective optimization problems (MaOPs) have four or more objectives to be optimized, where MOPs have at most three objectives. Many-objective optimization (MaOO) algorithms are used to find optimal trade-off solutions for MaOPs. As the number of objectives increases from MOPs to MaOPs the objective space also increases. This, in turn, makes it challenging to find truly optimal solutions that are also diverse. Another obstacle encountered with many-objectives is that the frequently used Pareto-dominance relation, used to define optimality for MOPs, degrades as the number of objectives increases [77, 106, 141]. Automotive engine calibration, for example, can easily be a ten-objective MaOP [104]. Imagine being the engineer responsible for the engine calibration of Lewis Hamilton's Formula One race car. Finding a set of optimal trade-off solutions for the engine parameters is vital to achieving success. There are also other real-world MaOO applications such as industrial scheduling [150, 172] and hybrid car controller optimization [111] to only name a few. There is, therefore, an urgent need for research to further develop effective MaOO algorithms as the number of MaOPs present in real life continues to increase [107].

Recently, a multi-objective multi-swarm variant of particle swarm optimization (PSO) [86], named multi-guide particle swarm optimization (MGPSO) [136], has been developed. Previous research results indicate that the MGPSO algorithm is highly competitive when compared against current PSO-based MOO algorithms as well as state-of-the-art multi-objective evolutionary algorithms (MOEAs) [136]. The MGPSO algorithm has also been shown to be suitable for efficiently solving MOPs [136]. However, no research has been done on the scalability of the MGPSO algorithm to solve MaOPs, requiring the simultaneous optimization of four or more objectives. It is expected that the MGPSO algorithm would not scale effectively to MaOPs because of the underlying dominance-relation being used that degrades as the number of objectives increases [77, 106, 141]. As mentioned, MaOPs are often encountered in real-world situations. Thus, to ensure real-world applicability it is highly desirable that MOO algorithms scale well. Therefore, this study not only aims to investigate the scalability of the MGPSO algorithm but also proposes two novel mechanisms that independently aim to improve the scalability of the MGPSO algorithm.

Therefore, the purpose of this thesis is first to investigate the ability of the MGPSO algorithm to scale with respect to the number of objectives for MaOPs of varying difficulty and complexity. The secondary purpose of this thesis is to propose and implement mechanisms that will help to improve the scalability of the MGPSO algorithm; that is, allowing the MGPSO algorithm to scale and effectively solve MaOPs. The proposed mechanisms will address the scalability limitations associated with the MGPSO algorithm as related to the number of objectives for a problem. This study does not consider scalability concerning the number of decision variables. This study assumes a static search space; that is, where the number of objectives remains fixed throughout the optimization. This study also assumes that each objective remains static (unchanged) throughout the search process. This study also considers only problems with boundary constraints.

1.2 Research Objectives

The main objective of this research is to implement mechanisms to allow the MGPSO algorithm to scale and effectively solve MaOPs. In light of the above, the following research objectives have been identified:

- To identify and discuss the mechanisms that promote scalability for the investigated state-of-the-art algorithms in effectively solving MaOPs.

- To investigate the ability of the MGPSO algorithm to solve MaOPs (i.e. scalability) by statistically comparing how well the MGPSO algorithm scales in comparison with other state-of-the-art MaOO algorithms on a set of benchmark problems.
- To propose two mechanisms (partial-dominance and knee-points) to help the MGPSO algorithm to scale; that is, being able to effectively solve MaOPs.
- To implement the two proposed mechanisms; that is, implement the two MGPSO algorithm adaptations, the partial-dominance multi-guide particle swarm optimization (PMGPSO) algorithm and the knee-point driven multi-guide particle swarm optimization (KnMGPSO) algorithm.
- To statistically compare the performance of the PMGPSO algorithm and the KnMGPSO algorithm with that of the original MGPSO algorithm and other state-of-the-art MaOO algorithms on a set of benchmark problems.
- To statistically compare the performance of the PMGPSO algorithm with that of the KnMGPSO algorithm on a set of benchmark problems, in order to determine if any of the two approaches (partial-dominance or knee-points) has best performance.
- As a sub-objective, this study also aims to investigate the use of different dynamic archive balance coefficient update strategies for the purpose of aiding the scalability of the MGPSO algorithm (as well as the PMGPSO and the KnMGPSO algorithms).

1.3 Contributions

The main contributions of this study are:

- The finding that the original MGPSO does scale well to MaOPs considering the hypervolume (HV) rankings.
- The finding that the original MGPSO does not scale well to MaOPs considering the inverted generational distance (IGD) rankings.
- The proposal of the PMGPSO algorithm, which uses partial-dominance to help improve the scalability of the original MGPSO algorithm.
- The finding that the PMGPSO algorithm scales competitively in terms of IGD.
- The proposal of the KnMGPSO algorithm, which uses knee-points to help improve the scalability of the original MGPSO algorithm.

- The finding that the KnMGPSO algorithm scales competitively in terms of IGD.
- The proposal of using a dynamic archive balance coefficient update strategy to further promote algorithm scalability.
- The finding that the investigated dynamic archive balance coefficient update strategies did not improve the scalability of the MGPSO, the PMGPSO, or the KnMGPSO algorithms.
- The finding that the MGPSO, the PMGPSO, and the KnMGPSO algorithms perform competitively relative to the other state-of-the-art MaOO benchmark algorithms.
- The finding that resampling control parameter values for the MGPSO, PMGPSO, and the KnMGPSO algorithms from the convergent regions is an effective method to avoid control parameter tuning while still ensuring competitive algorithm performance.
- The finding that equal subswarm sizes were sufficient.
- The finding that the suggested values used for the desired ratio of knee-points to non-dominated solutions were sufficient for the KnMGPSO algorithm.
- The finding that the multi-swarm approach, employed by the MGPSO algorithm and its adaptations, may play an important role in the scalability of MaOO algorithms.
- The finding that no algorithm outperformed any other algorithm in all cases considering both performance measures.

1.4 Dissertation Outline

The remainder of the dissertation proceeds as follows:

- **Chapter 2** discusses the relevant background information including MOO, MaOO, test problems, and performance measures.
- **Chapter 3** continues with relevant background information related to MaOO algorithms which include evolutionary algorithms and PSO approaches. This chapter also gives an overview of the algorithms used in this study.
- **Chapter 4** proposes the partial-dominance approach as related to aiding the scalability of the MGPSO algorithm and presents the PMGPSO algorithm. This chapter also discusses the empirical process that was

followed throughout this study to compare the various algorithms. The use of different dynamic archive balance coefficient update strategies for the original MGPSO algorithm and the PMGPSO algorithm is also investigated with the hope of improving scalability. Finally, the results are presented and discussed.

- **Chapter 5** proposes the knee-points approach as related to aiding the scalability of the MGPSO algorithm and presents the KnMGPSO algorithm. The use of different dynamic archive balance coefficient update strategies for the purpose of improving the scalability of the KnMGPSO algorithm is also investigated. Finally, the results are presented and discussed.

- **Chapter 6** presents and discusses the experimental results for the PMGPSO and KnMGPSO algorithms when compared in isolation. That is, to determine if either is best.

- **Chapter 7** concludes the dissertation with a summary of all the findings and conclusions of the presented work. This chapter also suggests ideas for future research based on the presented work.

The dissertation is supplemented by several appendices listed below:

- **Appendix A** lists and defines the most important acronyms that are used or introduced throughout the dissertation.

- **Appendix B** lists and defines the mathematical symbols used throughout the dissertation.

- **Appendix C** provides the parameter configurations for the algorithms used throughout this study.

- **Appendix D** presents the raw results (inverted generational distance and hypervolume performance measure values) obtained by the algorithms on the different problem instances for each independent sample associated with Chapter 4.

- **Appendix E** presents the raw results (inverted generational distance and hypervolume performance measure values) obtained by the algorithms on the different problem instances for each independent sample associated with Chapter 5.

- **Appendix F** presents the raw results (inverted generational distance and hypervolume performance measure values) obtained by the algorithms on the different problem instances for each independent sample associated with Chapter 6.

- **Appendix G** lists all the publications derived from this work.

Note that the dissertation makes extensive use of colour figures and is best read in colour.

Chapter 2

Many-objective Optimization

"Artificial Intelligence is whatever hasn't been done yet."
— Larry Tesler*, the computer scientist behind cut, copy, and paste.*

"Any sufficiently advanced technology is indistinguishable from magic."
— Arthur C. Clarke

In a nutshell, an optimization algorithm is a search method that aims to find a solution to an optimization problem, such that a given quantity is optimized (minimized or maximized), possibly subject to a set of constraints. The fundamental components of an optimization problem, according to Engelbrecht [47], are briefly described below:

- An **objective function**, representing the quantity to be optimized. Let f denote the objective function. Note that a maximum of f is equivalent to a minimum of $-f$. Problems such as constraint satisfaction problems (CSPs) do not explicitly define an objective function but rather the objective is to find a solution that satisfies a set of constraints. However, CSPs are not considered in this investigation.

- A **set of unknowns/variables**, directly affecting the value of the objective function. Let $\mathbf{x}$ denote the (independent) variables. By implication, $f(\mathbf{x})$ quantifies the quality of the candidate solution $\mathbf{x}$.

- A **set of constraints**, restricting the possible values that may be assigned to the variables. Usually, and in this study, at least a set of boundary constraints are defined, which define the domain (range) of valid values for each variable. However, it is possible to have more complex constraints that limit which candidate solutions can be considered as solutions. Note that these more complex constraints are not considered for this work.

Furthermore, Engelbrecht [47] classified optimization problems based on several attributes including:

- The **number of variables**: univariate (one variable) or multivariate (more than one variable).
- The **type of variables**: continuous-valued, integer-valued, mixed-valued (combination of continuous- and integer-valued variables), or combinatorial (permutations of integer-valued variables).
- The **degree of nonlinearity of the objective function**: linear or non-linear.
- The **constraints used**: unconstrained or constrained.
- The **number of optima**: unimodal (one optimal solution), multimodal (more than one optimum), or deceptive (false optima). There are three types of optima including a global minimum/maximum, a strong local minimum/maximum, and a weak local minimum/maximum.
- The **number of optimization criteria**: uni/single-objective (one objective function), multi-objective (two or three objective functions), or many-objective (more than three objective functions). In the case of MOPs and MaOPs, the objectives have to be optimized simultaneously.

In light of the above, this work only considers boundary constrained continuous-valued multivariate multi- and many-objective static optimization problems in a static search space. Boundary constrained continuous-valued single-objective optimization can informally be described as the process of systematically choosing input values from within an allowed range and computing the value of the real-valued objective function with the goal of minimizing (or maximizing) the objective function value while satisfying the boundary constraints. The function represents the optimization problem. For the remainder of this dissertation, unless otherwise stated, optimization problem minimization is assumed since there is no loss of generality between minimization and maximization. This is due to the duality principle [27] which allows maximization objectives, $f'(\mathbf{x})$, to be rewritten as minimization objectives, $f(\mathbf{x})$, where $f(\mathbf{x}) = -f'(\mathbf{x})$. A formal definition of a single-objective optimization problem as well as single-objective optimization algorithms, and a few single-objective benchmark problems can be found in [47].

The rest of this chapter is organized as follows. Sections 2.1 and 2.2 formally define and discuss aspects related to multi- and many-objective optimization. This is followed by section 2.3 and section 2.4 which discuss and motivate the test problems and performance measures used in this study. Finally, this chapter is summarized in section 2.5.

2.1 Multi-objective Optimization

This section focusses on MOO and related concepts. Section 2.1.1 formally defines a MOP and other important concepts used throughout this study. Sections 2.1.2 and 2.1.3 discuss different approaches used to define optimality for MOPs.

2.1.1 Multi-objective Problem

Let $\mathcal{S} \subseteq \mathbb{R}^n$ denote the n-dimensional search space (also known as the decision space), and $\mathcal{F} \subseteq \mathcal{S}$ the feasible space as determined by the constraints. If there are only boundary constraints and the search space is defined by those boundary constraints, the feasible space is equal to the search space; that is, $\mathcal{F} = \mathcal{S}$. $\mathcal{O} \subseteq \mathbb{R}^{n_m}$ denotes the n_m-dimensional objective space. Let n and n_m denote to the number of decision variables and the number of objectives respectively. A MOP objective function translates a decision vector, $\mathbf{x} \in \mathcal{F}$, to an objective vector, $\mathbf{f}(\mathbf{x})$, such that $\mathbf{f}(\mathbf{x}) \in \mathcal{O}$. The decision and objective vectors are formally defined below.

Definition 2.1. Decision vector *A decision vector,* $\mathbf{x} = (x_1, x_2, ..., x_n) \in \mathcal{F}$, *is an n-dimensional vector representing the chosen values for an optimization problem.*

Definition 2.2. Objective vector *An objective* vector, $\mathbf{f}(\mathbf{x}) = (f(\mathbf{x})_1, f(\mathbf{x})_2, ..., f(\mathbf{x})_{n_m}) \in \mathcal{O}$, *is an n_m-dimensional vector representing the possible solutions for an optimization problem.*

Figure 2.1 illustrates a decision space, $\mathcal{F}$, its corresponding objective space, $\mathcal{O}$, along with a three-dimensional decision vector, $\mathbf{x}$, and the corresponding two-dimensional objective vector, $\mathbf{f}(\mathbf{x})$.

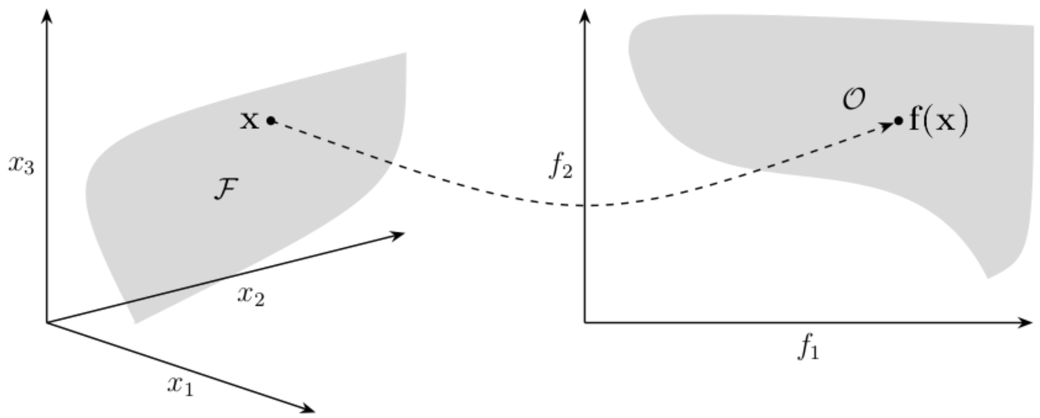


Figure 2.1: Decision and objective space.

Definition 2.3. Boundary constrained continuous-valued multi-objective optimization problem *A boundary constrained continuous-valued multi-objective optimization problem,* $\mathbf{f}(\mathbf{x})$*, with* n_m *(two or three) objectives is of the form:*

$$minimize \quad \mathbf{f}(\mathbf{x}) \quad = \quad (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_{n_m}(\mathbf{x})) \tag{2.1}$$

with $\mathbf{x} \in \mathcal{F} = \mathcal{S}$, $f_m : \mathbb{R}^n \rightarrow \mathbb{R}$, $\forall\, m \in \{1, ..., n_m\}$, $\mathcal{F} \subseteq \mathcal{S} \subset \mathbb{R}^n$ *is the feasible space as determined by the boundary constraints (*$\mathbf{x} \in [x_{min}, x_{max}]^n$*),* $\mathcal{S}$ *is the search space,* f_m *is the m-th objective function, and n is the number of dimensions or decision variables. Note for a MOP the number of objectives,* n_m*, can only be two or three.*

The interested reader is referred to [26, 107] where numerous benchmark, as well as real-life MOPs, are discussed. For MOO and MaOO, the definition of optimality is different than compared to uni-objective optimization (UOO). For UOO, where only one objective is optimized, the goal is to find a minimum or maximum for only that one objective function. For MOO or MaOO, the goal is to simultaneously find the minimum or maximum for several functions that are usually in conflict with one another. Consider the engine calibration example from Chapter 1. Assume, at the most basic level, that the goal of engine calibration is to only optimize the fuel consumption and acceleration. Intuitively, fuel consumption should ideally be minimized and acceleration should be maximized. Less fuel is cheaper and more acceleration leads to better race times. However, the amount of consumed fuel influences acceleration and vice versa. Therefore, a trade-off between these two objectives exists, and the goal is to find an optimal compromise. This example may be trivial but sufficiently demonstrates the trade-off(s) that exist within optimization problems with more than one objective. However, many real-world optimization problems require the optimization of MaOPs that are by definition more complex and not as easy to solve as simpler MOPs. Thus, optimal in terms of MOO and MaOO refers to finding a set of trade-off solutions that balance the opposing objectives. Sections 2.1.2 and 2.1.3 discuss different ways of mathematically describing this desired balance.

2.1.2 Weighted Aggregation Approach

The simplest way to define optimality for MOPs is the weighted aggregation approach [117, 118]. This approach defines an aggregate objective function as a weighted sum of the objectives. This approach has the advantage of enabling primitive UOO algorithms, without any modification, to solve MOPs since a single aggregate objective function encapsulates all the objective functions. However, using this approach also comes with some drawbacks including:

- The UOO algorithm has to be applied several times to find different solutions and even then there is no guarantee that repeated execution of the algorithm would yield different solutions. Alternatively, a niching (multi-solution) [5, 97, 98, 126] strategy could be used to obtain multiple solutions.

- The weight values for the weighted sum of objectives are problem dependent. Therefore, the weight values need to be optimized for each problem. There are methods for dynamically adjusting the weights such as random weight distribution, bang-bang weighted aggregation, and dynamic weighted aggregation [47, 83, 117, 118]. However, these methods are limited to only two-objective problems [47, 83, 117, 118].

- The Pareto-optimal front (POF) for a MOP is a set of solutions where each solution cannot improve any objective without degrading at least one of the other objectives. Weighted aggregation can only be used when the POF is concave regardless of the weight values [30, 83].

The advantages do not outweigh the disadvantages of this approach. Therefore, the weighted aggregation approach was not used in this study, but rather the Pareto-optimality approach discussed next.

2.1.3 Pareto-optimality Approach

Another way to define optimality for MOPs is the well-known Pareto-optimality approach, which was also used in this study. In order to ensure understanding and consistency when discussing Pareto-optimality, a number of frequently used definitions are listed below. Note that problem minimization is assumed.

Definition 2.4. Pareto-dominance *A decision vector* $\mathbf{x}_1 \in \mathcal{F}$ *dominates a decision* vector $\mathbf{x}_2 \in \mathcal{F}$ *(denoted by* $\mathbf{x}_1 \prec \mathbf{x}_2$*) if and only if* $f_m(\mathbf{x}_1) \leq f_m(\mathbf{x}_2)$ $\forall$ $m \in \{1, ..., n_m\}$ *and* $\exists$ $m \in \{1, ..., n_m\}$ *such that* $f_m(\mathbf{x}_1) < f_m(\mathbf{x}_2)$.

The concept of Pareto-dominance is illustrated in figure 2.2 for a two-objective MOP, $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}))$. The striped area denotes the area of objective vectors dominated by $\mathbf{f}(\mathbf{x})$.

Definition 2.5. Pareto-optimal *A decision vector* $\mathbf{x}_1 \in \mathcal{F}$ *is said to be Pareto-optimal if no decision vector* $\mathbf{x}_2 \in \mathcal{F}$ *exists such that* $\mathbf{x}_2 \prec \mathbf{x}_1$.

Definition 2.6. Pareto-optimal set *A set* $\mathcal{P} \subseteq \mathcal{F} \in \mathbb{R}^n$ *is said to be the Pareto-optimal set (POS) if it contains only Pareto-optimal decision vectors.* $\mathcal{P}$ *is formally defined as*

$$\mathcal{P} = \{\mathbf{x}_1 \in \mathcal{F} \,|\, \nexists\, \mathbf{x}_2 \in \mathcal{F} : \mathbf{x}_2 \prec \mathbf{x}_1\} \tag{2.2}$$

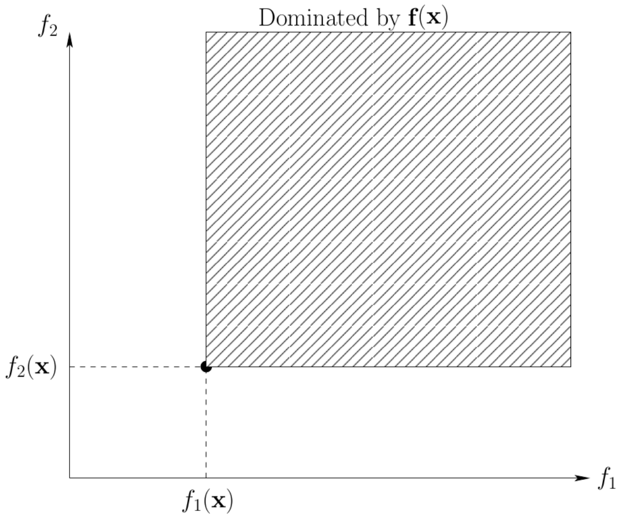


Figure 2.2: Illustration of Pareto-dominance.

Definition 2.7. Pareto-optimal Front *A set $\mathcal{Q} \subseteq \mathbb{R}^{n_m}$ is said to be the POF if it contains only objective vectors for Pareto-optimal decision vectors. $\mathcal{Q}$ is formally defined as*

$$\mathcal{Q} = \{f(\mathbf{x}) \,|\, \mathbf{x} \in \mathcal{P}\} \tag{2.3}$$

Figure 2.3 depicts a Pareto-optimal front along with the corresponding objective space.

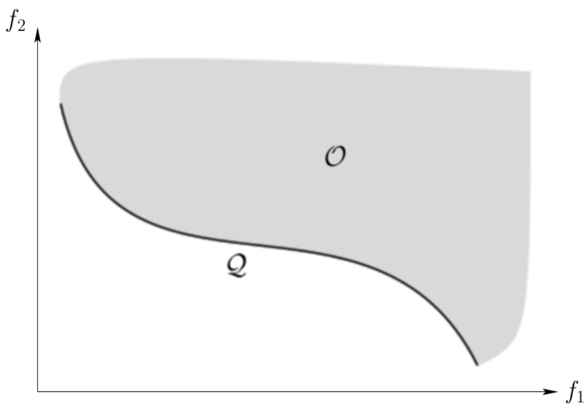


Figure 2.3: Pareto-optimal front.

Definition 2.8. Ideal objective vector *The ideal objective vector, $\mathbf{z}^*$, is a vector with components consisting of the optimal objective values for each of the n_m objective functions. $\mathbf{z}^*$ is formally defined as*

$$\mathbf{z}^* = (f_1^*, f_2^*, ..., f_{n_m}^*) \tag{2.4}$$

where f_m^* *is the optimal objective value for objective* $m \in \{1, ..., n_m\}$.

Definition 2.9. Nadir objective vector *The nadir objective vector,* $\mathbf{z}^{\text{nad}}$*, is a vector with components consisting of the worst objective values in the Pareto-optimal set,* $\mathcal{P}$*, for each of the* n_m *objective functions.* $\mathbf{z}^{\text{nad}}$ *is formally defined as*

$$\mathbf{z}^{\text{nad}} = (f_1^{**}, f_2^{**}, ..., f_{n_m}^{**}) \tag{2.5}$$

where $f_m^{**} = f_m(\mathbf{x}_1)$ *with* $\mathbf{x}_1 \in \mathcal{P} \,|\, \nexists\, \mathbf{x}_2 \in \mathcal{P} : f_m(\mathbf{x}_2) > f_m(\mathbf{x}_1)$.

Figure 2.4 depicts a Pareto-optimal front along with the ideal and nadir objective vectors.

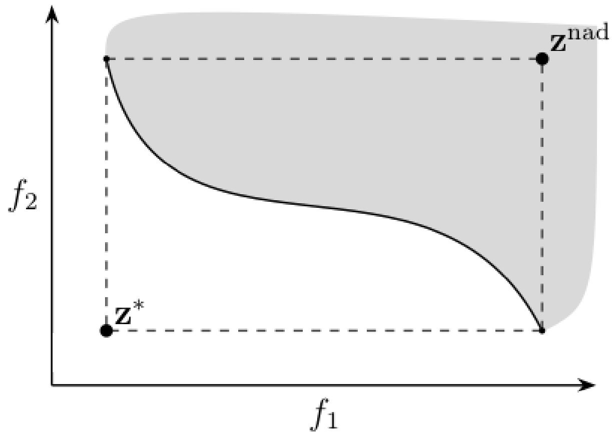


Figure 2.4: Illustration of Ideal ($\mathbf{z}^*$) and Nadir ($\mathbf{z}^{\text{nad}}$) objective vectors.

It is computationally expensive to find the true POF in the feasible decision space. Therefore, the goal of a MOO algorithm is to approximate the true POF and then to return the set of solutions representing the optimal compromises (i.e. the found POF). To effectively solve a MOP (or MaOP) a MOO (or MaOO) algorithm has the following objectives (which in itself is a MOP) [34, 178]:

- Minimize the distance from the approximated POF (i.e. the found solutions) to the true POF; that is, minimize inaccurate approximation.
- Maximize the diversity of the found front; that is, maximize solution spread. This implies that a MOP also has to maintain the already found non-dominated solutions.

2.2 Many-objective Optimization

This section focusses on MaOO and related concepts. More specifically, section 2.2.1 formally defines a MaOP and section 2.2.2 discusses the challenges related to MaOPs.

2.2.1 Many-objective Problem

In literature there is a distinction made between MOPs and MaOPs. That is, MOPs consist of only two or three objectives per problem whereas MaOPs have four or more objectives per problem. The reason for making this distinction between MOPs and MaOPs is probably rooted in the fact that optimization problems become increasingly more difficult to solve as the number of objectives continues to increase, therefore, scientists put MaOPs in a class of their own.

Note that the definition of a MOP, as defined in Equation (2.3), is essentially the same as that of a MaOP except that the latter has no restrictions on the maximum number of objectives while the former may have at most three objectives.

2.2.2 Many-objective Optimization Challenges

MaOPs are not as easily solved as MOPs. This is due to the following challenges faced by a MaOO algorithm when solving a MaOP:

- The frequently used Pareto-optimality concept, defined in section 2.1.3, breaks down as the number of objectives increases [77, 106, 141]. More specifically, the Pareto-dominance approach struggles to successfully identify good solutions for MaOPs since the majority of the candidate solutions are non-dominated early on in the search process, which results in the MaOO algorithm degrading to a random search, likely to find sub-optimal solutions [81]. Previous research shows that over 90% of a randomly generated set of initial solutions is non-dominated for eight or more objectives [68].

- Balancing solution diversity (exploration) and solution convergence (exploitation) become more difficult as the number of objectives increases [141]. This increased difficulty is due to the objective space that increases together with the number of objectives. Consequently, solutions found throughout the search will probably be distant from each other in the objective space [38], for example, when a MOEA chooses parents to mate and reproduce, the offspring will likely be far from the parents resulting in reduced algorithm effectiveness. Another consequence is that algorithms frequently terminate with a set of undesirable solutions that is either well-spread, but far from the true POF (diverse but lacking accuracy), or has converged to a small subregion close the true POF (accurate but lacking diversity). In other words, MaOO algorithms often terminate with a set of well-distributed non-dominated solutions, which are far from Pareto-optimal, or with a set of similar non-dominated nearly Pareto-optimal solutions.

- The crowding distance [35] value of a solution provides an estimate of the density of solutions surrounding that solution. Crowding distance is often used as a diversity preservation measure when maintaining already found non-dominated solutions during the search process. Crowding distance is also utilized to enhance the exploration of a search algorithm, i.e., a diversity promotion mechanism. Unfortunately, the crowding distance operator degrades as the number of objectives increases [91]. The crowding distance operator degrades due to favouritism towards dominance-resistant solutions [77, 105]. Dominance-resistant solutions have great performance in one objective and terrible performance in many others but are hardly ever dominated. These dominant-resistant solutions can misguide the MaOO algorithm to sub-optimal solutions. Therefore, maintaining a set of well-spread solutions, i.e. diversity, becomes more difficult. Note that dominant-resistant solutions should not be disregarded completely as they can promote solution diversity [77].

- Evaluation of diversity measures become computationally expensive. For example, calculation of the crowding distance for solutions is computationally expensive. Approximations can be made to alleviate this issue, but at the cost of possibly sacrificing the solution spread of the final found POF due to inaccuracies introduced by the approximations [106].

- Occasionally, computationally expensive calculations, such as the hypervolume (HV) [181] calculation, are required to measure algorithm performance. HV becomes exponentially more computationally expensive with an increase in the number of objectives [168] and thus becomes practically infeasible to calculate when the number of objectives is large. The Monte Carlo sampling technique [4] is often used to estimate the HV for MaOPs [95, 106, 174]. Unfortunately, this will result in some degree of inaccuracy when comparing MaOO algorithms.

- The number of solutions required to approximate the entire POF increases exponentially as the number of objectives increases [105].

- Selecting a final solution from the approximated POF at the end of the search (upon algorithm termination) is challenging due to the difficulty in visualizing high-dimensional trade-off surfaces [106]. This challenge relates more to the practical applications of MaOO.

Note that the same approach for defining optimality for MOPs as discussed in section 2.1.3 also applies to MaOPs. A MaOO algorithm also has the same *raison d'être*[1] as that of a MOO algorithm; that is, to find a set of diverse and accurate solutions.

Taking into consideration the above mentioned many-objective challenges, a MaOO algorithm that only incorporates the Pareto-dominance approach to define optimality usually terminates with an unsatisfactory set of solutions. The Pareto-optimality concept is also used by the original MGPSO algorithm [106]. Therefore, this study proposes and implements mechanisms to investigate if the scalability of the MGPSO algorithm can be improved. That is, these mechanisms aim to improve the ability of the MGPSO algorithm to effectively solve MaOPs while adhering to the goals of a MaOO algorithm as described in section 2.1.3.

A number of potential solutions for scaling to MaOPs (i.e. MaOO approaches) have been proposed in previous literature to remedy the challenges above. These approaches can largely be separated into six distinct categories which include:

- **Pareto-dominance relation modification:** These approaches modify the Pareto-dominance relation to promote convergence to the POF [33, 44, 55, 64, 68, 93, 131, 132, 166, 182].

- **Secondary convergence metric:** These approaches employ a secondary convergence-related metric alongside the Pareto-dominance relation [90, 96, 102, 105, 174].

- **Dimensionality reduction:** These approaches identify and discard the least non-conflicting objectives that can be removed without changing the POS [14, 15, 99, 103, 108, 134, 148]. For example, dismissing objectives that are highly correlated with others.

- **Performance measure integration:** These approaches integrate one or more performance metrics which implicitly take diversity and convergence into consideration [4, 10, 179]. The largest downfall to indicator-based optimizers is the computational overhead that is required to calculate useful indicators, e.g. HV. Note that some progress has already been made in terms of reducing the computational complexity of performance measures [4, 13, 58, 167].

- **Decomposition:** These approaches use decomposition to decompose a MaOP problem into a set of subproblems to be optimized simultaneously [38, 95, 173].

- **Reference- or preference-based:** Interactive user preferences [39] or reference-points [38] approaches have also been proposed.

[1] reason for being

2.3 Test Problems

To properly investigate a MaOO algorithm, it is crucial to have a set of well-understood problems on which to test the algorithm. This set of test problems (also known as the benchmark problems, the benchmark functions, the test suite, or the benchmark suite) needs to include problems that can challenge the ability of a MaOO algorithm to achieve its two primary objectives; that is, solution convergence and solution diversity. To accurately and thoroughly evaluate a MaOO algorithm, the benchmark problems should have several properties [34]. Properties that challenge convergence include multimodality, deception, and isolated optima [136]. Properties that challenge diversity include convexity or non-convexity, discreteness, and non-uniformity [136].

There exist several benchmark suites for single-, multi-, and many-objective optimization problems [107]. The two problem sets chosen for this work present a mix of different desirable properties useful for properly testing MaOO algorithms. The two chosen benchmark problem sets are presented and discussed in sections 2.3.1 and 2.3.2. Most importantly, the chosen benchmark functions are scalable both in terms of the number of objectives and the number of decision variables. Note, however, that this study only considers scalability concerning the number of objectives.

2.3.1 Deb-Thiele-Laumanns-Zitzler Test Problems

Deb *et al.* [41] proposed a benchmark suite, referred to as the Deb-Thiele-Laumanns-Zitzler (DTLZ) test suite. This set consists of nine scalable problems, DTLZ1 through DTLZ9. However, two of these are constrained optimization problems (DTLZ8 and DTLZ9) and, therefore, are omitted from this study. Note that for all the problems defined below n_m refers to the number of objectives and n refers to the number of decision variables. A set Z of n parameters are divided into two distinct sets as follows:

$$\begin{aligned}
\text{Given } Z &= \{z_1, ..., z_n\} &&= \{z_1, ..., z_j, z_{j+1}, ..., z_n\} \\
\text{let } P &= \{p_1, ..., p_l\} &&= \{z_1, ..., z_j\} \\
D &= \{d_1, ..., d_k\} &&= \{z_{j+1}, ..., z_n\}
\end{aligned}$$

where P is a set of j position parameters, D is a set of k distance parameters, and the total number of parameters is $n = j + k$. For this study k was defined as $k = n - n_m + 1$. Note that for all the DTLZ problems defined below $z_i \in [0, 1], \forall\, i = 1, 2, ..., n$. DTLZ1 through DTLZ7 are defined as follows (all objectives are to be minimized):

DTLZ1

$$\begin{aligned}
f_1(Y) &= \big(1+g(D)\big)\times 0.5\times \textstyle\prod_{i=1}^{n_m-1} p_i \\
f_{m=2:n_m-1}(P) &= \big(1+g(D)\big)\times 0.5\times (\textstyle\prod_{i=1}^{n_m-m} p_i)(1-p_{n_m-m+1}) \\
f_{n_m}(P) &= \big(1+g(D)\big)\times 0.5\times (1-p_1) \\
\text{where } g(D) &= 100\Big[k+\textstyle\sum_{i=1}^{k}\big((d_i-0.5)^2-\cos(20\pi(d_i-0.5))\big)\Big]
\end{aligned} \tag{2.6}$$

Using $D = \mathbf{0}$ will result in the Pareto-optimal solutions with objective function values on the linear hyperplane $\sum_{m=1}^{n_m} f_m = 0.5$ [41]. The search space contains $(11^k - 1)$ local optima making it difficult to convergence to the hyperplane [41]. The problem can be made more complex by using other difficult multi-modal g functions (using a larger k) and/or replacing d_i by non-linear mapping as described in [41].

DTLZ2

$$\begin{aligned}
f_1(P) &= \big(1+g(D)\big)\textstyle\prod_{i=1}^{n_m-1}\cos(p_i\pi/2) \\
f_{m=2:n_m-1}(P) &= \big(1+g(D)\big)\big(\textstyle\prod_{i=1}^{n_m-m}\cos(p_i\pi/2)\big)\sin(p_{n_m-m+1}\pi/2) \\
f_{n_m}(P) &= \big(1+g(D)\big)\sin(p_1\pi/2) \\
\text{where } g(D) &= \textstyle\sum_{i=1}^{k}(d_i-0.5)^2
\end{aligned} \tag{2.7}$$

The Pareto-optimal solutions are obtained with $D = \mathbf{0.5}$ which will result in objective function values that satisfy $\sum_{i=1}^{n_m} f_i^2 = 1$ [41]. To make the problem more difficult, each variable z_i for $i = 1$ to $(n_m - 1)$ can be replaced by the mean value of the variables as described in [41].

DTLZ3

$$\begin{aligned}
f_1(P) &= \big(1+g(D)\big)\textstyle\prod_{i=1}^{n_m-1}\cos(p_i\pi/2) \\
f_{m=2:n_m-1}(P) &= \big(1+g(D)\big)\big(\textstyle\prod_{i=1}^{n_m-m}\cos(p_i\pi/2)\big)\sin(p_{n_m-m+1}\pi/2) \\
f_{n_m}(P) &= \big(1+g(D)\big)\sin(p_1\pi/2) \\
\text{where } g(D) &= 100\Big[k+\textstyle\sum_{i=1}^{k}\big((d_i-0.5)^2-\cos(20\pi(d_i-0.5))\big)\Big]
\end{aligned} \tag{2.8}$$

Note that DTLZ3 is similar to DTLZ2 except for the equation for g which is replaced by the one from DTLZ1. There are $(3^k - 1)$ local optima and one global optimum [41]. All local POFs (local optima) are parallel to the global POF (global optimum); that is, the global POF is at $g = 0$ and the next local optimum is at $g = 1$ [41]. The global POF corresponds to $D = \mathbf{0.5}$ [41]. The problem can be made more difficult by using a large k or a higher-frequency cosine function [41].

DTLZ4

$$\begin{aligned} f_1(P) &= \big(1+g(D)\big)\textstyle\prod_{i=1}^{n_m-1}\cos(p_i^{100}\pi/2) \\ f_{m=2:n_m-1}(P) &= \big(1+g(D)\big)\big(\textstyle\prod_{i=1}^{n_m-m}\cos(p_i^{100}\pi/2)\big)\sin(p_{n_m-m+1}^{100}\pi/2) \\ f_{n_m}(P) &= \big(1+g(D)\big)\sin(p_1^{100}\pi/2) \\ \text{where } g(D) &= \textstyle\sum_{i=1}^{k}(d_i-0.5)^2 \end{aligned} \tag{2.9}$$

DTLZ4 is another modification of DTLZ2, which induces a bias towards the $f_{n_m} - f_1$ plane; that is, allowing a dense collection of solutions to exist near this plane [41].

DTLZ5

$$\begin{aligned} f_1(P) &= \big(1+g(D)\big)\textstyle\prod_{i=1}^{n_m-1}\cos(\theta_i\pi/2) \\ f_{m=2:n_m-1}(P) &= \big(1+g(D)\big)\big(\textstyle\prod_{i=1}^{n_m-m}\cos(\theta_i\pi/2)\big)\sin(\theta_{n_m-m+1}\pi/2) \\ f_{n_m}(P) &= \big(1+g(D)\big)\sin(\theta_1\pi/2) \\ \text{where } g(D) &= \textstyle\sum_{i=1}^{k}(d_i-0.5)^2 \\ \text{and } \theta_i &= \begin{cases} \frac{\pi}{4\left(1+g(D)\right)}\big(1+2g(D)p_i\big), & \text{for } i=2,3,\ldots,(n_m-1) \\ p_i, & \text{for } i=1 \end{cases} \end{aligned} \tag{2.10}$$

DTLZ5 is yet another modified version of DTLZ2 which introduces the θ-mapping ensuring that there is only one POF (one global optimum or non-dominated region) in the entire search space [41]. The global POF corresponds to $D = \mathbf{0.5}$ [26].

DTLZ6

$$\begin{aligned} f_1(P) &= \big(1+g(D)\big)\textstyle\prod_{i=1}^{n_m-1}\cos(\theta_i\pi/2) \\ f_{m=2:n_m-1}(P) &= \big(1+g(D)\big)\big(\textstyle\prod_{i=1}^{n_m-m}\cos(\theta_i\pi/2)\big)\sin(\theta_{n_m-m+1}\pi/2) \\ f_{n_m}(P) &= \big(1+g(D)\big)\sin(\theta_1\pi/2) \\ \text{where } g(D) &= \textstyle\sum_{i=1}^{k}d_i^{0.1} \\ \text{and } \theta_i &= \begin{cases} \frac{\pi}{4\left(1+g(D)\right)}\big(1+2g(D)p_i\big), & \text{for } i=2,3,\ldots,(n_m-1) \\ p_i, & \text{for } i=1 \end{cases} \end{aligned} \tag{2.11}$$

DTLZ6 is similar to DTLZ5 except for the modified g equation. The global POF corresponds to $D = \mathbf{0}$ [26].

DTLZ7

$$\begin{aligned} f_{m=1:n_m-1}(P) &= y_m \\ f_{n_m}(P) &= \big(1+g(D)\big)\Big(n_m - \textstyle\sum_{i=1}^{n_m-1}\Big[\frac{f_i(P)}{1+g(D)}\big(1+\sin(3\pi f_i(P))\big)\Big]\Big) \\ \text{where } g(D) &= 1+\tfrac{9}{k}\textstyle\sum_{i=1}^{k} d_i \end{aligned} \tag{2.12}$$

DTLZ7 has 2^{n_m-1} disconnected Pareto-optimal regions which will test the ability of an algorithm to maintain a subpopulation in each of these optimal regions of the search space [41]. The Pareto-optimal solutions are obtained by using $D = \mathbf{0}$ [41]. The problem can be made harder by using a higher-frequency sine function or using a multi-modal g function as described in [41].

Table 2.1 summarizes other important properties of DTLZ1 through DTLZ7. The POFs for the three-objective DTLZ1 through DTLZ7 problems are visualized from different viewing angles in figures 2.5 to 2.11 as taken from [26]. For more detail, about the DTLZ test suite, the reader is referred to [41].

Table 2.1: DTLZ Benchmark Function Properties

Name	**Objective**	**Separability**	**Bias**	**Geometry**	**Modality**
DTLZ1	$f_{1:n_m}$	Separable	No	Linear	Multi
DTLZ2	$f_{1:n_m}$	Separable	No	Concave	Uni
DTLZ3	$f_{1:n_m}$	Separable	No	Concave	Multi
DTLZ4	$f_{1:n_m}$	Separable	Yes	Concave	Uni
DTLZ5	$f_{1:n_m}$	Unknown	No	Unknown	Uni
DTLZ6	$f_{1:n_m}$	Unknown	Yes	Unknown	Uni
DTLZ7	$f_{1:n_m-1}$ f_{n_m}	Not applicable Separable	No	Disconnected	Uni Multi

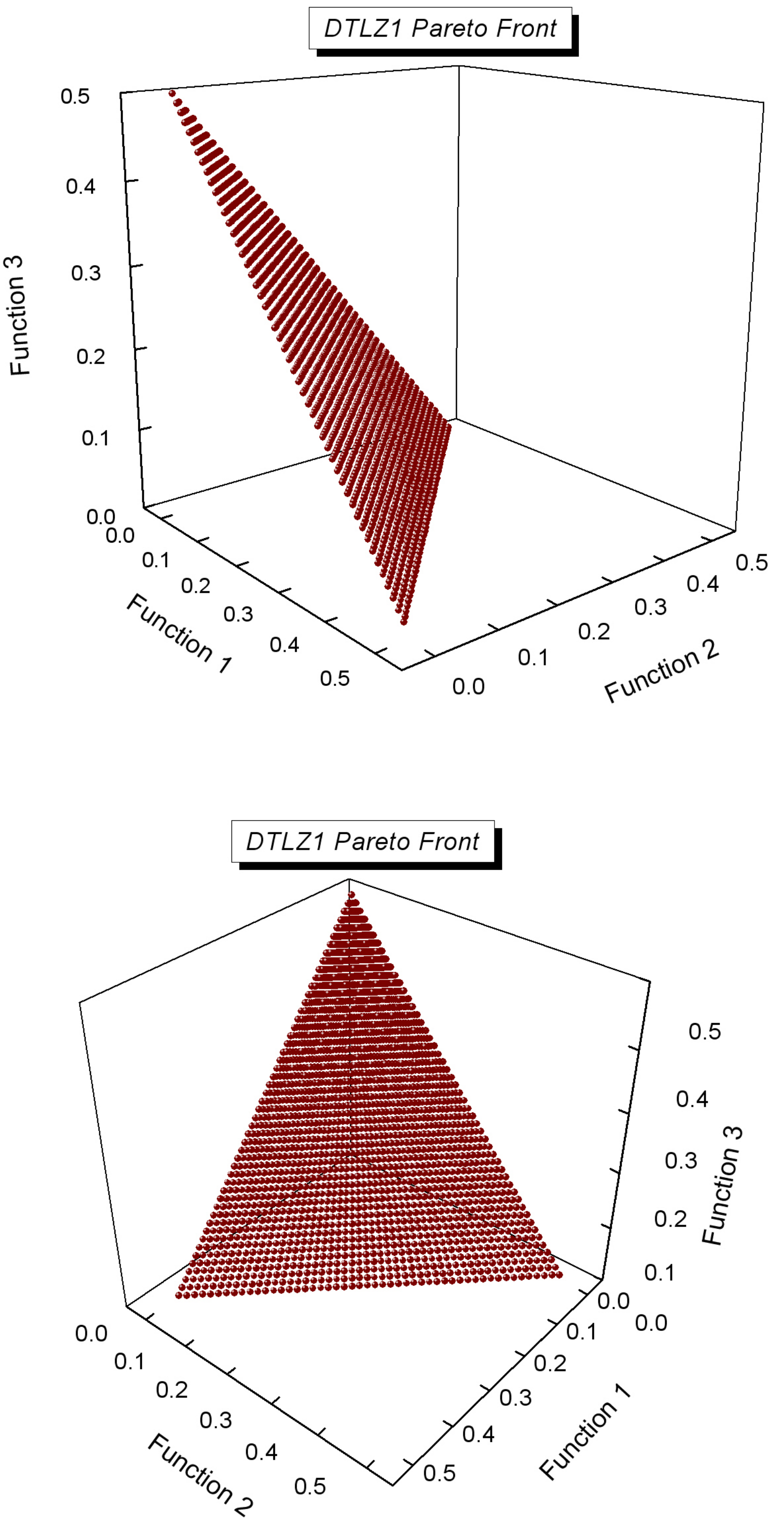


Figure 2.5: 3-objective DTLZ1 Pareto Front

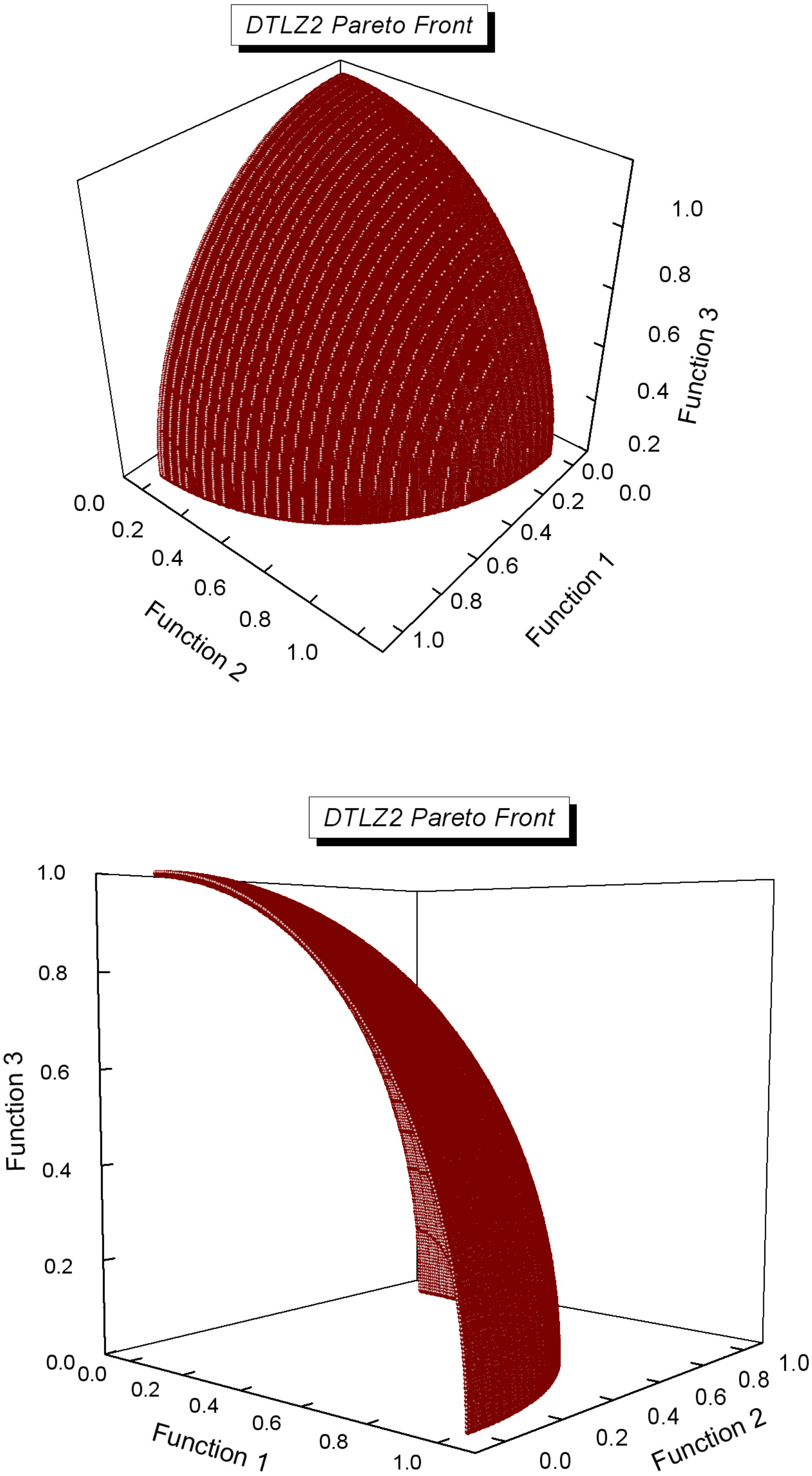


Figure 2.6: 3-objective DTLZ2 Pareto Front

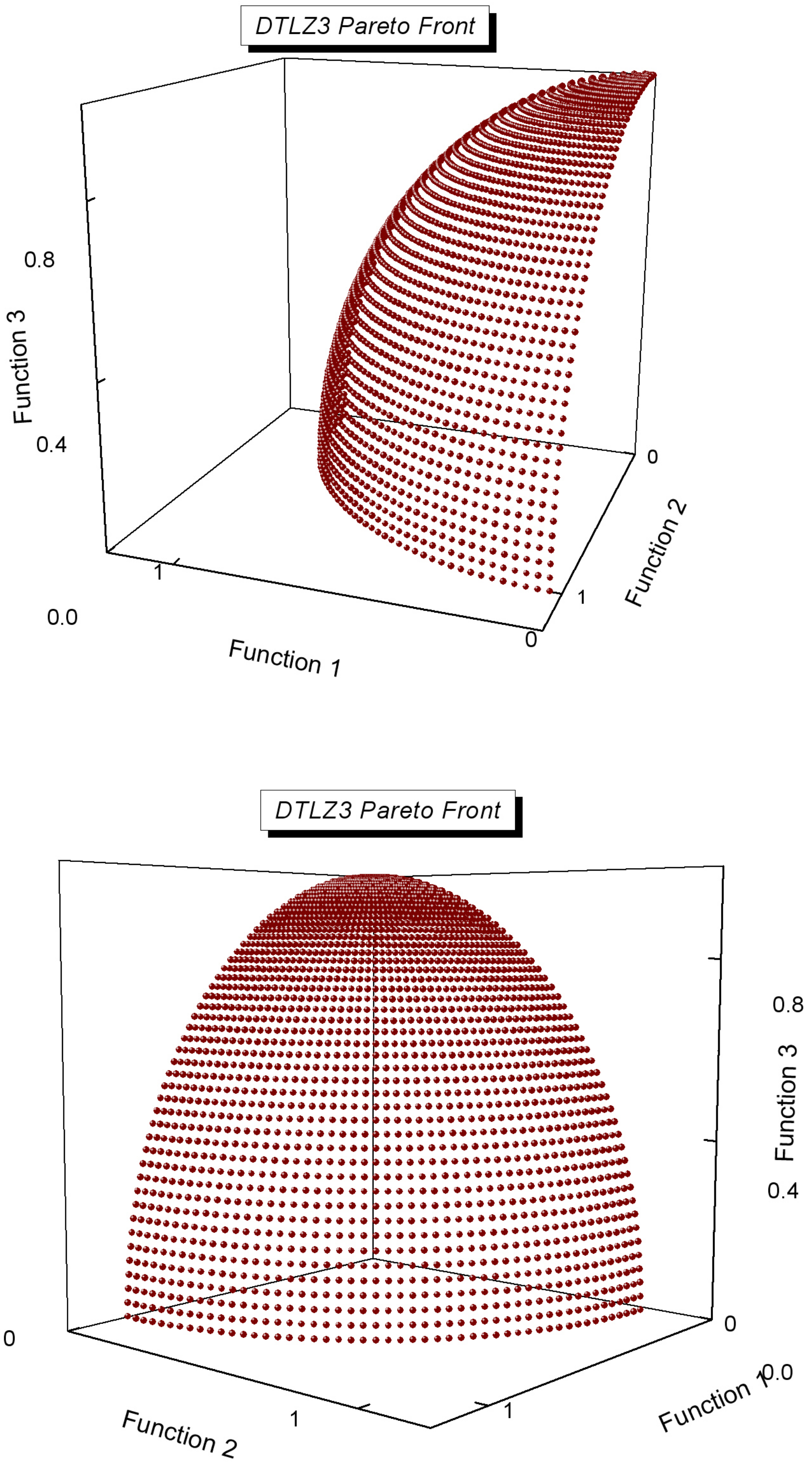


Figure 2.7: 3-objective DTLZ3 Pareto Front

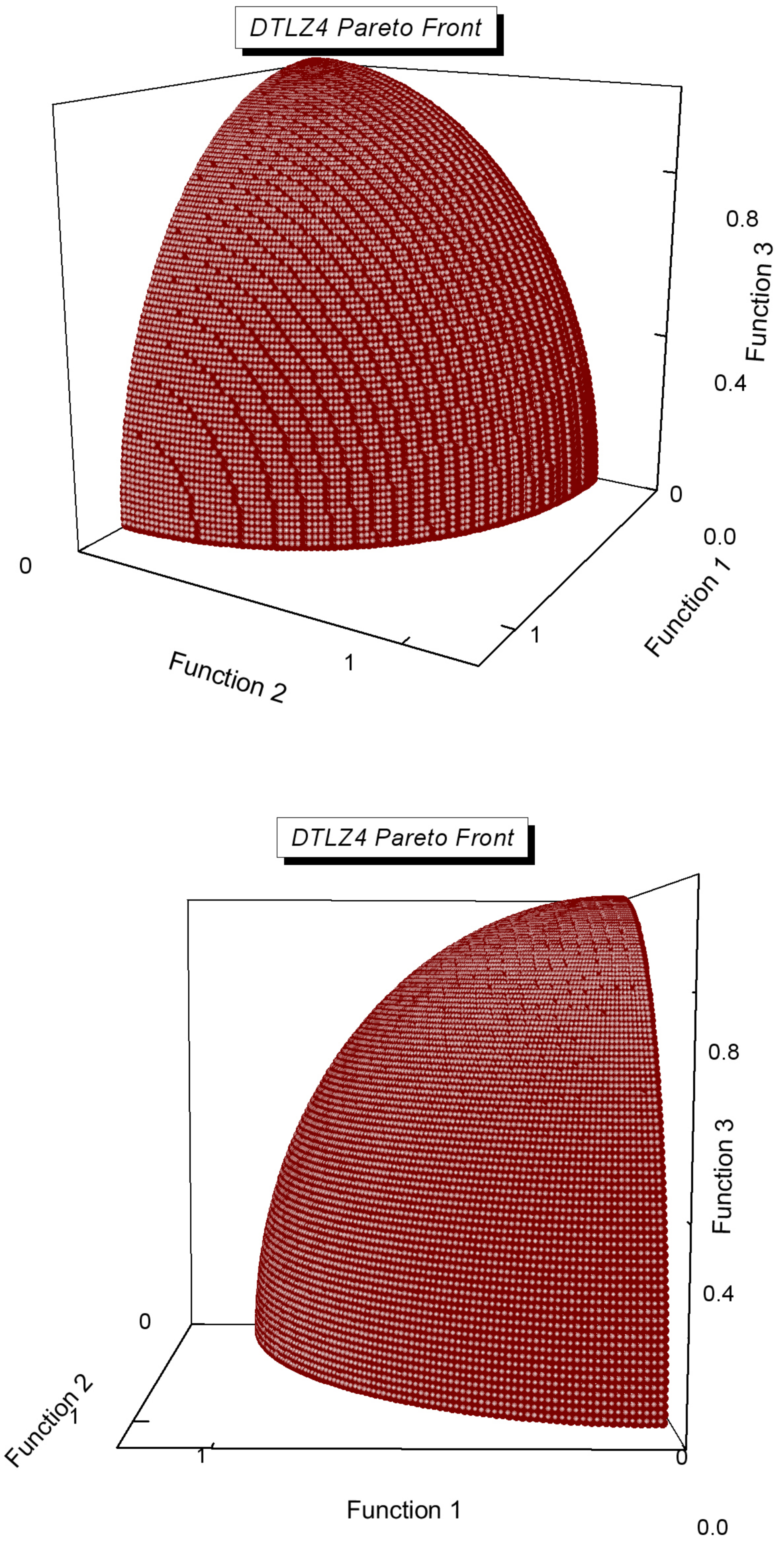
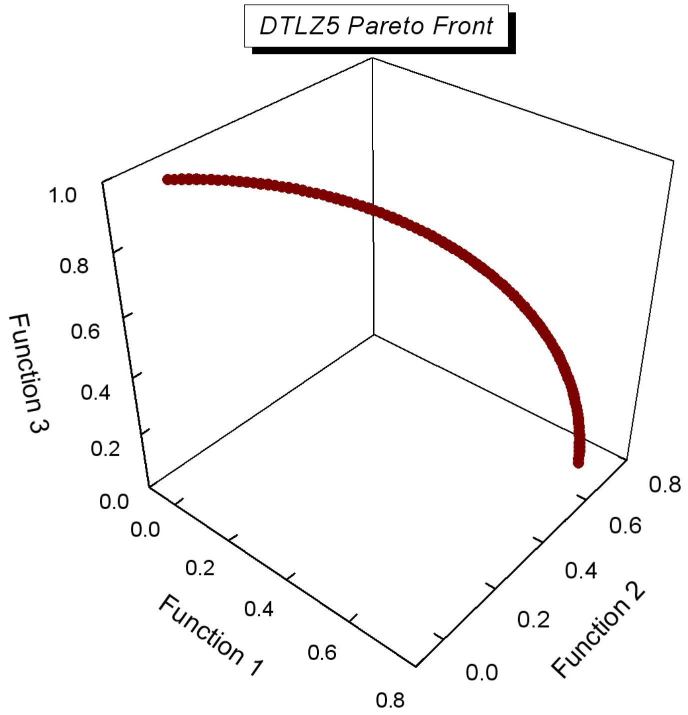


Figure 2.8: 3-objective DTLZ4 Pareto Front



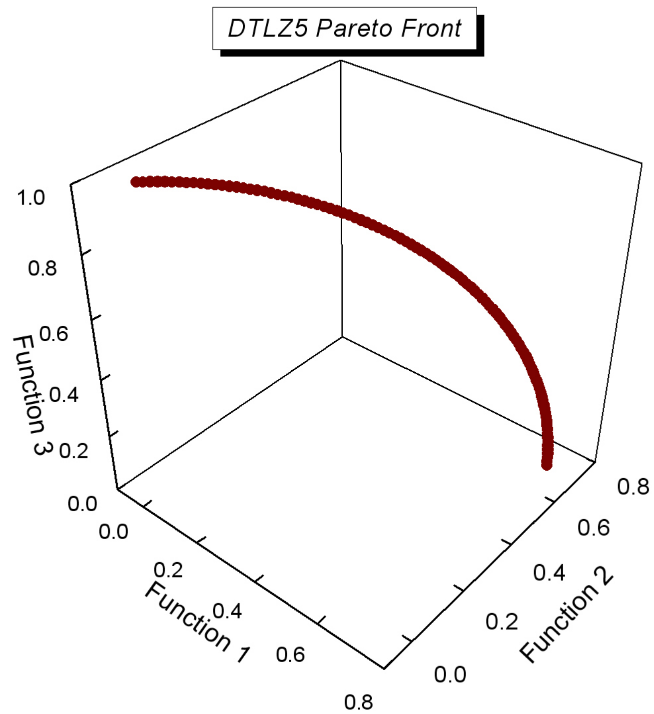


Figure 2.9: 3-objective DTLZ5 Pareto Front

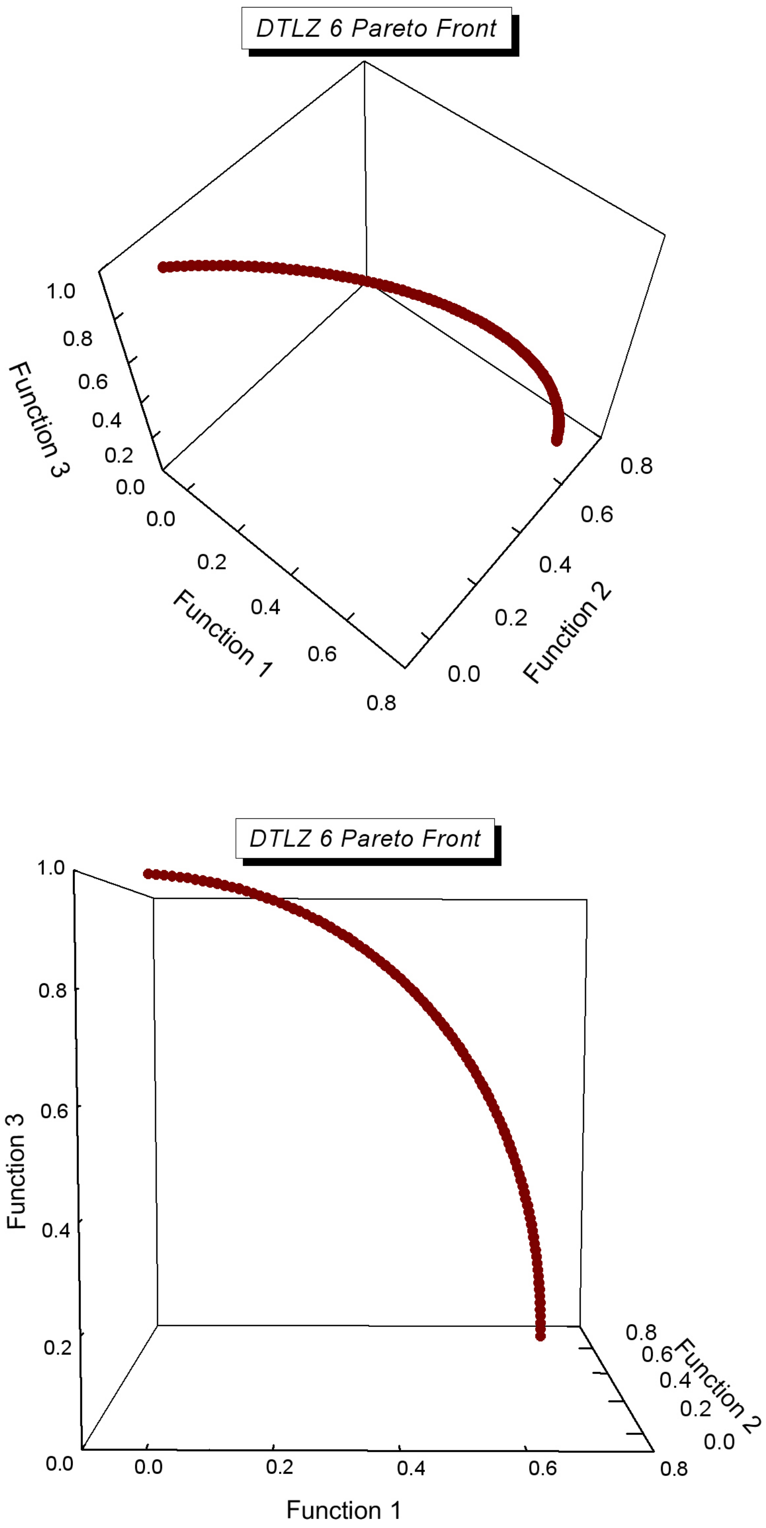


Figure 2.10: 3-objective DTLZ6 Pareto Front

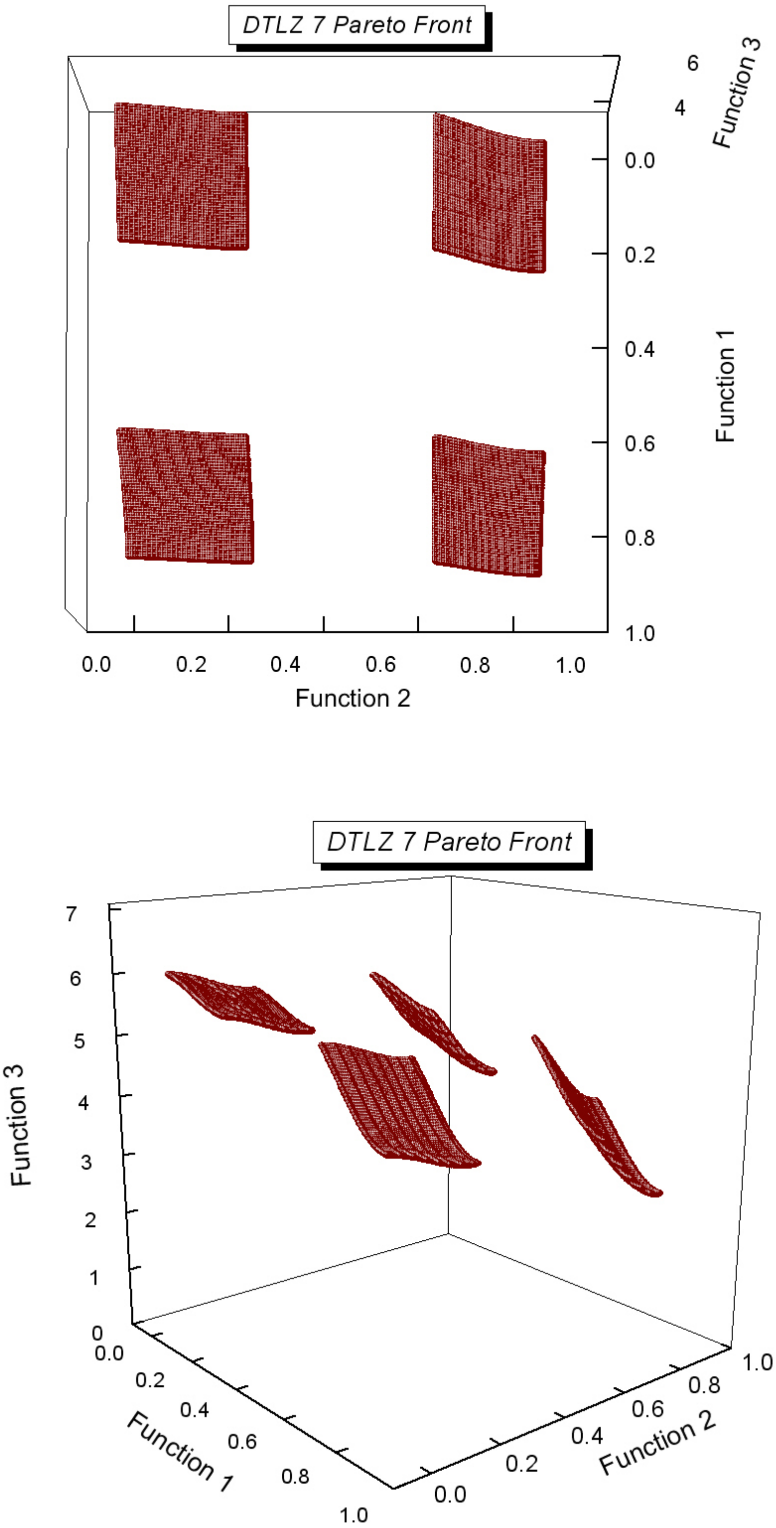


Figure 2.11: 3-objective DTLZ7 Pareto Front

2.3.2 Walking Fish Group Test Problems

Huband *et al.* [74, 75] proposed a benchmark suite, referred to as the Walking Fish Group (WFG) test suite. This set consists of nine scalable problems, WFG1 through WFG9. Minimization is assumed for all objectives. The WFG problems are of the following form:

$$
\begin{aligned}
\text{Given } Z &= \{z_1, ..., z_k, z_{k+1}, ..., z_n\} \\
\text{minimize } f_m(X) &= qx_{n_m} + s_m h_m(x_1, ..., x_{n_m-1}),\ \forall\, m \in \{1, ..., n_m\} \\
\text{where } X &= \{x_1, ..., x_{n_m}\} \\
&= \{max(t^p_{n_m}, o_1)(t^p_1 - 0.5) + 0.5, ..., \\
&\quad\ max(t^p_{n_m}, o_{n_m-1})(t^p_{n_m-1} - 0.5) + 0.5, t^p_{n_m}\} \\
\mathbf{t}^p &= \{t^p_1, ..., t^p_{n_m}\} \hookleftarrow \mathbf{t}^{p-1} \hookleftarrow ... \hookleftarrow \mathbf{t}^1 \hookleftarrow Z_{[0,1]} \\
Z_{[0,1]} &= \{z_{1,[0,1]}, ..., z_{n,[0,1]}\} \\
&= \left\{\frac{z_1}{z_{1,max}}, ..., \frac{z_n}{z_{n,max}}\right\}
\end{aligned}
\tag{2.13}
$$

where

- Z is a set of $k + l = n \geq n_m$ working parameters, where the first k working parameters are position-related parameters and the last l working parameters are distance-related parameters. For this study k was defined as $k = 2(n_m - 1)$. The domain for all $z_i \in Z$ is $[0, z_{i,max}]$, where all $z_{i,max} > 0$. Note that the domain of all $x_i \in X$ is $[0, 1]$;
- X is a set of n_m underlying parameters, where x_{n_m} is an underlying distance parameter and $x_{1:n_m-1}$ are underlying position parameters;
- $q > 0$ is a distance scaling constant;
- $s_{1:n_m} > 0$ are scaling constants;
- $o_{1:n_m-1} \in \{0, 1\}$ are degeneracy constants, and for each $o_i = 0$, the dimensionality of the POF is reduced by one;
- $h_{1:n_m}$ are shape functions; and
- $\mathbf{t}^{1:p}$ are transition vectors, where "$\hookleftarrow$" indicates that each transition vector is created from another vector via transformation functions.

The nature of the POF is determined by the shape functions. The shape functions map parameters with a domain $[0, 1]$ onto a range $[0, 1]$. The following shape functions are defined:

Linear

$$\begin{aligned}
\text{linear}_1(x_1,...,x_{n_m-1}) &= \textstyle\prod_{i=1}^{n_m-1} x_i \\
\text{linear}_{m=2:n_m-1}(x_1,...,x_{n_m-1}) &= (\textstyle\prod_{i=1}^{n_m-m} x_i)(1-x_{n_m-m+1}) \\
\text{linear}_{n_m}(x_1,...,x_{n_m-1}) &= 1-x_1
\end{aligned} \tag{2.14}$$

When $h_{m=1:n_m} = \text{linear}_m$, the POF is a linear hyperplane, where $\sum_{m=1}^{n_m} h_m = 1$ [74, 75].

Convex

$$\begin{aligned}
\text{convex}_1(x_1,...,x_{n_m-1}) &= \textstyle\prod_{i=1}^{n_m-1}\left(1-\cos(x_i\frac{\pi}{2})\right) \\
\text{convex}_{m=2:n_m-1}(x_1,...,x_{n_m-1}) &= \left(\textstyle\prod_{i=1}^{n_m-m}(1-\cos(x_i\frac{\pi}{2}))\right)\left(1-\sin(x_{n_m-m+1}\frac{\pi}{2})\right) \\
\text{convex}_{n_m}(x_1,...,x_{n_m-1}) &= 1-\sin(x_1\frac{\pi}{2})
\end{aligned} \tag{2.15}$$

When $h_{m=1:n_m} = \text{convex}_m$, the POF is purely convex [74, 75].

Concave

$$\begin{aligned}
\text{concave}_1(x_1,...,x_{n_m-1}) &= \textstyle\prod_{i=1}^{n_m-1}\sin(x_i\frac{\pi}{2}) \\
\text{concave}_{m=2:n_m-1}(x_1,...,x_{n_m-1}) &= \left(\textstyle\prod_{i=1}^{n_m-m}\sin(x_i\frac{\pi}{2})\right)\left(\cos(x_{n_m-m+1}\frac{\pi}{2})\right) \\
\text{concave}_{n_m}(x_1,...,x_{n_m-1}) &= \cos(x_1\frac{\pi}{2})
\end{aligned} \tag{2.16}$$

When $h_{m=1:n_m} = \text{concave}_m$, the POF is purely concave, and a region of the hypersphere of radius one is centred at the origin, where $\sum_{m=1}^{n_m} h_m^2 = 1$ [74, 75].

Mixed convex/concave ($\alpha > 0, a \in \{1,2,...\}$)

$$\text{mixed}_{n_m}(x_1,...,x_{n_m-1}) = (1-x_1-\frac{\cos(2a\pi x_1+\frac{\pi}{2})}{2a\pi})^{\alpha} \tag{2.17}$$

Equation (2.17) will result in a POF with both concave and convex segments. The number of convex and concave segments are controlled by a [74, 75]. The overall shape is controlled by α: when $\alpha > 1$, the overall shape is convex, when $\alpha < 1$, the overall shape is concave, and when $\alpha = 1$, the overall shape is linear [74, 75].

Disconnected ($\alpha, \beta > 0, a \in \{1,2,...\}$)

$$\text{disc}_{n_m}(x_1,...,x_{n_m-1}) = 1-(x_1)^{\alpha}\cos^2\left(a(x_1)^{\beta}\pi\right) \tag{2.18}$$

Equation (2.18) causes the POF to have disconnected regions. The number of disconnected regions is controlled by a [74, 75]. The overall shape is controlled by α: when $\alpha > 1$, the overall shape is convex, when $\alpha < 1$, the overall shape is concave, and when $\alpha = 1$, the overall shape is linear [74, 75]. The location of the disconnected regions is controlled by β: larger values of β push the location of the disconnected regions towards larger values of x_1, and vice versa [74, 75].

The purpose of transformation functions is to map input parameters with domain [0, 1] onto the range [0, 1]. The following transformation functions have been defined:

Bias: Polynomial $(\alpha > 0, \alpha \neq 1)$

$$\text{b\_poly}(y, \alpha) = y^{\alpha} \tag{2.19}$$

When $\alpha > 1$, y is biased towards zero, and when $\alpha < 1$, y is biased towards one [74, 75].

Bias: Flat Region $(a, b, c \in [0, 1], b < c, b = 0 \Rightarrow a = 0 \wedge c \neq 1, c = 1 \Rightarrow a = 1 \wedge b \neq 0)$

$$\begin{aligned} \text{b\_flat}(y, a, b, c) &= a + min\big(0, \lfloor y - b \rfloor\big)\tfrac{a(b-y)}{b} \\ &\quad - min\big(0, \lfloor c - y \rfloor\big)\tfrac{(1-a)(y-c)}{1-c} \end{aligned} \tag{2.20}$$

Values y between b and c (i.e. the area of the flat region) are all mapped to the value of a [74, 75].

Bias: Parameter Dependent $(a \in (0, 1), 0 < b < c)$

$$\begin{aligned} \text{b\_param}(y, \mathbf{y}'a, b, c) &= y^{b+(c-b)v(u(\mathbf{y}'))} \\ \text{where } v(u(\mathbf{y}')) &= a - \big(1 - 2u(\mathbf{y}')\big)\big|\lfloor 0.5 - u(\mathbf{y}') \rfloor + a\big| \end{aligned} \tag{2.21}$$

a, b, c, and the secondary parameter vector $\mathbf{y}'$ together determine the extent to which y is biased by being raised to some power, values of $u(\mathbf{y}\prime) \in [0, 0.5]$ are mapped linearly onto $[b, b + (c - b)a]$, and values of $u(\mathbf{y}\prime) \in [0.5, 1]$ are mapped linearly onto $[b + (c - b)a, c]$ [74, 75].

Shift: Linear $(a \in (0, 1))$

$$\text{s\_linear}(y, a) = \frac{y-a}{\big|\lfloor a-y \rfloor + a\big|} \tag{2.22}$$

a is the value for which y is mapped to zero [74, 75].

Shift: Deceptive $(a \in (0, 1), 0 < b \ll 1, 0 < c \ll 1, a - b > 0, a + b < 1)$

$$\begin{aligned} \text{s\_decept}(y, a, b, c) &= 1 + (|y - a| - b) \times \\ &\quad \left(\frac{\lfloor y-a+b \rfloor (1 - C + \frac{a-b}{b})}{a-b} + \frac{\lfloor a+b-y \rfloor (1 - c + \frac{1-a-b}{b})}{1-a-b} + \frac{1}{b} \right) \end{aligned} \tag{2.23}$$

a is the value for which y is mapped to zero, and the global minimum of the transformation [74, 75]. The size of the well/basin "opening" (leading to the global minimum at a) is b, and c is the value of the only two deceptive minima [74, 75].

Shift: Multimodal $(a \in \{1, 2, ...\}, b \geq 0, (4a+2)\pi \geq 4b, c \in (0, 1))$

$$\text{s\_multi}(y, a, b, c) = \frac{1+\cos\left[(4a+2)\pi\left(0.5-\frac{|y-c|}{2(\lfloor c-y \rfloor+c)}\right)\right]+4b\left(\frac{|y-c|}{2(\lfloor c-y \rfloor+c)}\right)^2}{b+2} \tag{2.24}$$

The number of optima is controlled by a, the magnitude of the "hill sizes" of the multi-modality is controlled by b, and c is the value for which y is mapped to zero [74, 75]. When $b = 0$, $2a+1$ values of y (one at c) are mapped to zero, and when $b \neq 0$, there are $2a$ local minima, and one global minimum at c [74, 75]. More difficult problems can be created with larger values of a and smaller values of B [74, 75].

Reduction: Weighted Sum $(|\mathbf{w}| = |\mathbf{y}|, w_1, ..., w_{|y|} > 0)$

$$\text{r\_sum}(\mathbf{y}, \mathbf{w}) = \frac{\sum_{i=1}^{|\mathbf{y}|} w_i y_i}{\sum_{i=1}^{|\mathbf{y}|} w_i} \tag{2.25}$$

The constant weight vector $\mathbf{w}$ forces an algorithm to treat the parameter vector, $\mathbf{y}$, differently [74, 75]. For conciseness, a weighted product reduction function, equivalent to the weighted sum reduction function defined above, has been omitted.

Reduction: Non-separable $(a \in \{1, ..., |\mathbf{y}|\}, |\mathbf{y}| \bmod a = 0)$

$$\text{r\_nonsep}(\mathbf{y}, a) = \frac{\sum_{j=1}^{|\mathbf{Y}|}\left(y_j + \sum_{k=0}^{a-2} |y_j - y_{1+(j+k) \bmod |\mathbf{y}|}|\right)}{\frac{|\mathbf{y}|}{a}\lceil \frac{a}{2} \rceil (1+2a-2\lceil \frac{a}{2} \rceil)} \tag{2.26}$$

The degree of non-separability is controlled by a [74, 75], noting that

$$\text{r\_nonsep}(\mathbf{y}, 1) = \text{r\_sum}(\mathbf{y}, \mathbf{1})$$

The role of the bias transformations is to bias the fitness landscape [74, 75]. The location of optima is moved by shift transformations (subject to skewing by bias transformations) [74, 75]. The deceptive and multimodal shift transformations make the corresponding problem deceptive and multimodal, respectively [74, 75]. The flat region transformation can also have a significant impact on the fitness landscape [74, 75].

The following restrictions apply to ensure that the problems have a well-balanced design:

Constants
Constants must be fixed values and cannot be tied to the value of any parameter [74, 75].

Primary Parameters
For any given transition vector, all the parameters of the originating transition vector must be employed exactly once as a primary parameter, counting parameters that appear independently as primary parameters, and in the same order in which the parameters appear in the originating transition vector [74, 75].

Secondary Parameters
Care must be taken to avoid cyclical dependencies in b_param [74, 75]. Consider the following terminology: if $param_1$ is a primary parameter of b_param, and $param_2$ is a secondary parameter, then $param_1$ depends on $param_2$. If $param_2$ likewise depends on $param_3$, then $param_1$ depends, indirectly, on $param_3$ [74, 75]. To prevent cyclical dependencies, no two parameters should be dependent on one another. A parameter should also not depend on itself.

Shifts
Parameters should only be subjected to a maximum of one shift transformation [74, 75].

Reductions
Reduction transformations should belong to transition vectors that are closer to the underlying parameter vector than any shift transformations [74, 75].

b_flat
When $a = 0$, b_flat should only belong to transition vectors that are further away from the underlying parameter vector than any shift or reduction transformation [74, 75].

The constant values and domains of the working parameters for the nine WFG test problems are defined as follows:

Constants

$$\begin{aligned} s_{m=1:n_m} &= 2m \\ o_1 &= 1 \\ o_{2:n_m-1} &= \begin{cases} 0 \text{ for WFG3} \\ 1 \text{ otherwise} \end{cases} \end{aligned} \tag{2.27}$$

The settings for $s_{1:n_m}$ ensure that the POFs have dissimilar trade-off magnitudes, and the settings for $o_{1:n_m-1}$ ensure that the POFs are not degenerate, except in the case of WFG3, which has a one-dimensional POF [74, 75].

Domains

$$z_{i=1:n,max} = 2i \tag{2.28}$$

The working parameters have domains of dissimilar magnitude [74, 75].

Finally, the nine WFG test problems can be defined by using a combination of the shape, transformation functions, and restrictions as presented above. WFG1 through WFG9 are defined as follows:

WFG1

$$\begin{array}{llcl}
\text{Shape} & h_{m=1:n_m-1} & = & \text{convex}_m \\
 & h_{n_m} & = & \text{mixed}_{n_m}, \text{with } \alpha = 1 \text{ and } a = 5 \\
\mathbf{t}^1 & t^1_{i=1:k} & = & y_i \\
 & t^1_{i=k+1:n} & = & \text{s_linear}(y_i, 0.35) \\
\mathbf{t}^2 & t^2_{i=1:k} & = & y_i \\
 & t^2_{i=k+1:n} & = & \text{b_flat}(y_i, 0.8, 0.75, 0.85) \\
\mathbf{t}^3 & t^3_{i=1:n} & = & \text{b_poly}(y_i, 0.02) \\
\mathbf{t}^4 & t^4_{i=1:n_m-1} & = & \text{r_sum}\big((y_{(i-1)k/(n_m-1)+1}, ..., y_{ik/(n_m-1)}), \\
 & & & (2((i-1)k/(n_m-1)+1), ..., 2ik/(n_m-1))\big) \\
 & t^4_{n_m} & = & \text{r_sum}\big((y_{k+1}, ..., y_n), (2(k+1), ..., 2n)\big)
\end{array} \tag{2.29}$$

WFG2

$$\begin{array}{llcl}
\text{Shape} & h_{m=1:n_m-1} & = & \text{convex}_m \\
 & h_{n_m} & = & \text{disc}_{n_m}, \text{with } \alpha = \beta = 1 \text{ and } a = 5 \\
\mathbf{t}^1 & t^1_{i=1:k} & = & y_i \\
 & t^1_{i=k+1:n} & = & \text{s_linear}(y_i, 0.35) \\
\mathbf{t}^2 & t^2_{i=1:k} & = & y_i \\
 & t^2_{i=k+1:k+l/2} & = & \text{r_nonsep}\big((y_{k+2(i-k)-1}, y_{k+2(i-k)}), 2\big) \\
\mathbf{t}^3 & t^3_{i=1:n_m-1} & = & \text{r_sum}\big((y_{(i-1)k/(n_m-1)+1}, ..., y_{ik/(n_m-1)}), (1, ..., 1)\big) \\
 & t^3_{n_m} & = & \text{r_sum}\big((y_{k+1}, ..., y_{k+l/2}), (1, ..., 1)\big)
\end{array} \tag{2.30}$$

WFG3

$$
\begin{array}{rrcl}
\text{Shape} & h_{m=1:n_m} & = & \text{linear}_m \\
\mathbf{t}^1 & t^1_{i=1:k} & = & y_i \\
 & t^1_{i=k+1:n} & = & \text{s\_linear}(y_i, 0.35) \\
\mathbf{t}^2 & t^2_{i=1:k} & = & y_i \\
 & t^2_{i=k+1:k+l/2} & = & \text{r\_nonsep}\big((y_{k+2(i-k)-1}, y_{k+2(i-k)}), 2\big) \\
\mathbf{t}^3 & t^3_{i=1:n_m-1} & = & \text{r\_sum}\big((y_{(i-1)k/(n_m-1)+1}, ..., y_{ik/(n_m-1)}), (1, ..., 1)\big) \\
 & t^3_{n_m} & = & \text{r\_sum}\big((y_{k+1}, ..., y_{k+l/2}), (1, ..., 1)\big)
\end{array}
\tag{2.31}
$$

Note that the only difference between WFG2 and WFG3 is the shape function; that is, WFG2 uses a convex shape function whereas WFG3 uses a linear shape function.

WFG4

$$
\begin{array}{rrcl}
\text{Shape} & h_{m=1:n_m} & = & \text{concave}_m \\
\mathbf{t}^1 & t^1_{i=1:n} & = & \text{s\_multi}(y_i, 30, 10, 0.35) \\
\mathbf{t}^2 & t^2_{i=1:n_m-1} & = & \text{r\_sum}\big((y_{(i-1)k/(n_m-1)+1}, ..., y_{ik/(n_m-1)}), (1, ..., 1)\big) \\
 & t^2_{n_m} & = & \text{r\_sum}\big((y_{k+1}, ..., y_n), (1, ..., 1)\big)
\end{array}
\tag{2.32}
$$

WFG5

$$
\begin{array}{rrcl}
\text{Shape} & h_{m=1:n_m} & = & \text{concave}_m \\
\mathbf{t}^1 & t^1_{i=1:n} & = & \text{s\_decept}(y_i, 0.35, 0.001, 0.05) \\
\mathbf{t}^2 & t^2_{i=1:n_m-1} & = & \text{r\_sum}\big((y_{(i-1)k/(n_m-1)+1}, ..., y_{ik/(n_m-1)}), (1, ..., 1)\big) \\
 & t^2_{n_m} & = & \text{r\_sum}\big((y_{k+1}, ..., y_n), (1, ..., 1)\big)
\end{array}
\tag{2.33}
$$

Note that the only difference between WFG4 and WFG5 is the shift transformation function used; that is, WFG4 uses a multimodal shift whereas WFG5 uses a deceptive shift.

WFG6

$$\begin{array}{lrcl}
\text{Shape} & h_{m=1:n_m} & = & \text{concave}_m \\
\mathbf{t}^1 & t^1_{i=1:k} & = & y_i \\
 & t^1_{i=k+1:n} & = & \text{s_linear}(y_i, 0.35) \\
\mathbf{t}^2 & t^2_{i=1:n_m-1} & = & \text{r_nonsep}\big((y_{(i-1)k/(n_m-1)+1}, ..., y_{ik/(n_m-1)}), k/(n_m-1)\big) \\
 & t^2_{n_m} & = & \text{r_nonsep}\big((y_{k+1}, ..., y_n), l\big)
\end{array} \tag{2.34}$$

Note that WFG1 and WFG6 use the same shift transformation function; that is, a linear shift.

WFG7

$$\begin{array}{lrcl}
\text{Shape} & h_{m=1:n_m} & = & \text{concave}_m \\
\mathbf{t}^1 & t^1_{i=1:k} & = & \text{b_param}\big(y_i, \text{r_sum}((y_{i+1}, ..., y_n), (1, ..., 1)), \\
 & & & \frac{0.98}{49.98}, 0.02, 50\big) \\
 & t^1_{i=k+1:n} & = & y_i \\
\mathbf{t}^2 & t^2_{i=1:k} & = & y_i \\
 & \mathbf{t}^2_{i=k+1:n} & = & \text{s_linear}(y_i, 0.35) \\
\mathbf{t}^3 & t^3_{i=1:n_m-1} & = & \text{r_sum}\big((y_{(i-1)k/(n_m-1)+1}, ..., y_{ik/(n_m-1)}), (1, ..., 1)\big) \\
 & t^3_{n_m} & = & \text{r_sum}\big((y_{k+1}, ..., y_n), (1, ..., 1)\big)
\end{array} \tag{2.35}$$

Note that WFG7 uses the linear shift transformation like WFG1 and the weighted sum reduction like WFG4.

WFG8

$$\begin{array}{lrcl}
\text{Shape} & h_{m=1:n_m} & = & \text{concave}_m \\
\mathbf{t}^1 & t^1_{i=1:k} & = & y_i \\
 & t^1_{i=k+1:n} & = & \text{b_param}\big(y_i, \text{r_sum}((y_1, ..., y_{i-1}), (1, ..., 1)), \\
 & & & \frac{0.98}{49.98}, 0.02, 50\big) \\
\mathbf{t}^2 & t^2_{i=1:k} & = & y_i \\
 & t^2_{i=k+1:n} & = & \text{s_linear}(y_i, 0.35) \\
\mathbf{t}^3 & t^3_{i=1:n_m-1} & = & \text{r_sum}\big((y_{(i-1)k/(n_m-1)+1}, ..., y_{ik/(n_m-1)}), (1, ..., 1)\big) \\
 & t^3_{n_m} & = & \text{r_sum}\big((y_{k+1}, ..., y_n), (1, ..., 1)\big)
\end{array} \tag{2.36}$$

Note that both WFG8 and WFG7 use the same linear shift transformation and weighted sum reduction as WFG1 and WFG4 respectively.

WFG9

$$
\begin{array}{llcl}
\text{Shape} & h_{m=1:n_m} & = & \text{concave}_m \\
\mathbf{t}^1 & t^1_{i=1:n-1} & = & \text{b\_param}\big(y_i, \text{r\_sum}((y_{i+1}, ..., y_n), (1, ..., 1)), \\
 & & & \frac{0.98}{49.98}, 0.02, 50\big) \\
 & t^1_n & = & y_n \\
\mathbf{t}^2 & t^2_{i=1:k} & = & \text{s\_decep}(y_i, 0.35, 0.001, 0.05) \\
 & t^2_{i=k+1:n} & = & \text{s\_multi}(y_i, 30, 95, 0.35) \\
\mathbf{t}^3 & t^3_{i=1:n_m-1} & = & \text{r\_nonsep}\big((y_{(i-1)k/(n_m-1)+1}, ..., y_{ik/(n_m-1)}), k/(n_m-1)\big) \\
 & t^3_{n_m} & = & \text{r\_nonsep}\big((y_{k+1}, ..., y_n), l\big)
\end{array}
\tag{2.37}
$$

Note that WFG9 uses the same non-separable reduction function as WFG6.

The Pareto-optimal solutions for WFG1 to WFG9 are obtained when the following conditions are satisfied:

- For WFG1 to WFG7 a solution is Pareto-optimal iff all

$$z_{i=k+1:n} = 2i \times 0.35$$

- For WFG8 it is required that all of

$$
\begin{aligned}
z_{i=k+1:n} &= 2i \times 0.35^{\left(0.02+49.98\left(\frac{0.98}{49.98}-(1-2u)|\lfloor 0.5-u\rfloor+\frac{0.98}{49.98}|\right)\right)^{-1}} \\
\text{where } u &= \text{r\_sum}\big((z_1, ..., z_{i-1}), (1, ..., 1)\big)
\end{aligned}
$$

- For WFG9 it is required that all of

$$
\begin{aligned}
z_{i=k+1:n} &= 2i \times \begin{cases} 0.35^{(0.02+1.96u)^{-1}}, \ i \neq n \\ 0.35, \ i = n \end{cases} \\
\text{where } u &= \text{r\_sum}\big((z_{i+1}, ..., z_n), (1, ..., 1)\big)
\end{aligned}
$$

  which can be found by first determining z_n, then z_{n-1}, and so on, until the required value for z_{k+1} is determined [74, 75]. Once the optimal values for $z_{k+1:n}$ are determined, the position-related parameters can randomly be varied to obtain different Pareto-optimal solutions [74, 75].

Table 2.2 summarizes other important properties of WFG1 through WFG9. The POFs for three-objective WFG1 through WFG9 problems are visualized from different viewing angles in figures 2.12 to 2.20 as taken from [26]. For more detail, about the WFG test suite, the reader is referred to [74, 75].

Table 2.2: WFG Benchmark Function Properties

Name	**Objective**	**Separability**	**Bias**	**Geometry**	**Modality**
WFG1	$f_{1:n_m}$	Separable	Yes	Convex, Mixed	Uni
WFG2	$f_{1:n_m-1}$ f_{n_m}	Non-Separable	No	Convex, Disconnected	Uni Multi
WFG3	$f_{1:n_m}$	Non-Separable	No	Linear, Degenerate	Uni
WFG4	$f_{1:n_m}$	Separable	No	Concave	Multi
WFG5	$f_{1:n_m}$	Separable	No	Concave	Deceptive
WFG6	$f_{1:n_m}$	Non-Separable	No	Concave	Uni
WFG7	$f_{1:n_m}$	Separable	Yes	Concave	Uni
WFG8	$f_{1:n_m}$	Non-Separable	Yes	Concave	Uni
WFG9	$f_{1:n_m}$	Non-Separable	Yes	Concave	Multi, Deceptive

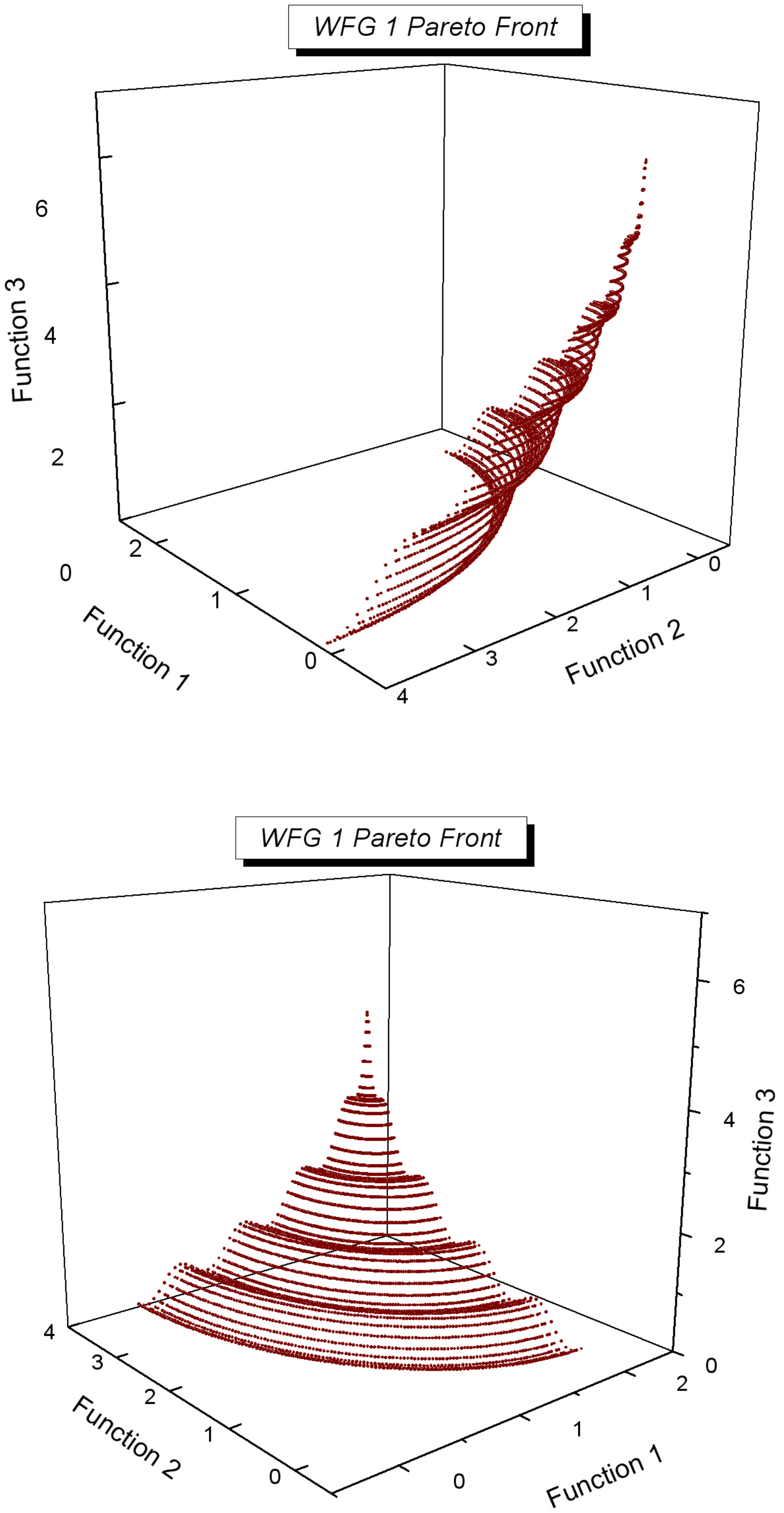
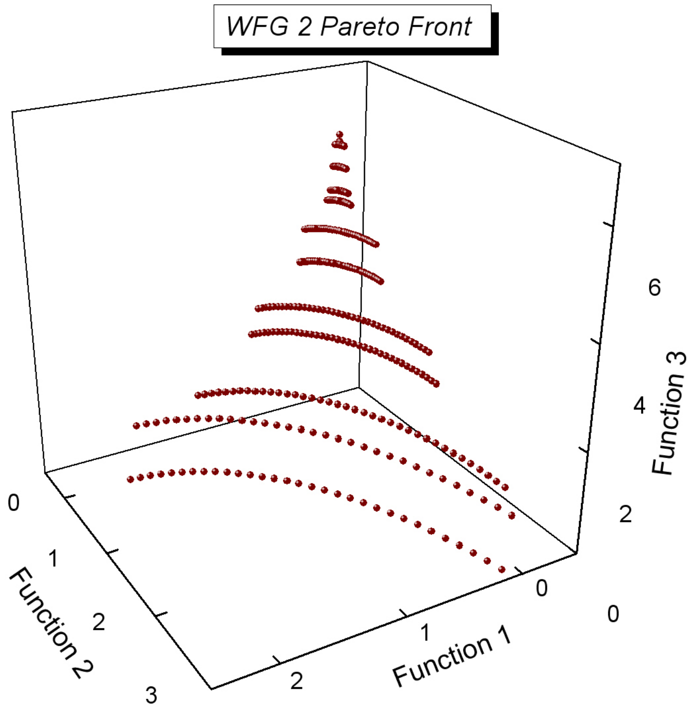


Figure 2.12: 3-objective WFG1 Pareto Front



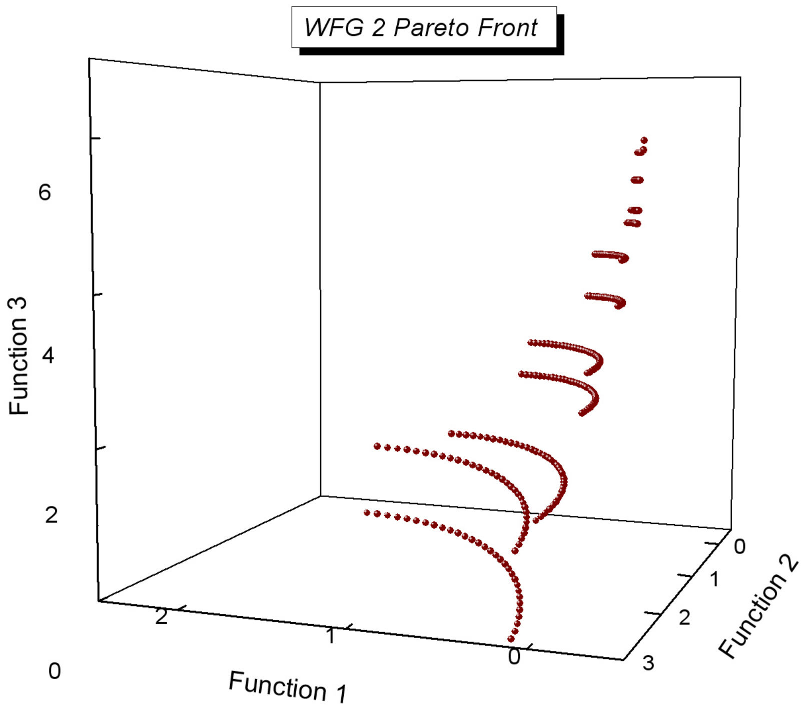


Figure 2.13: 3-objective WFG2 Pareto Front

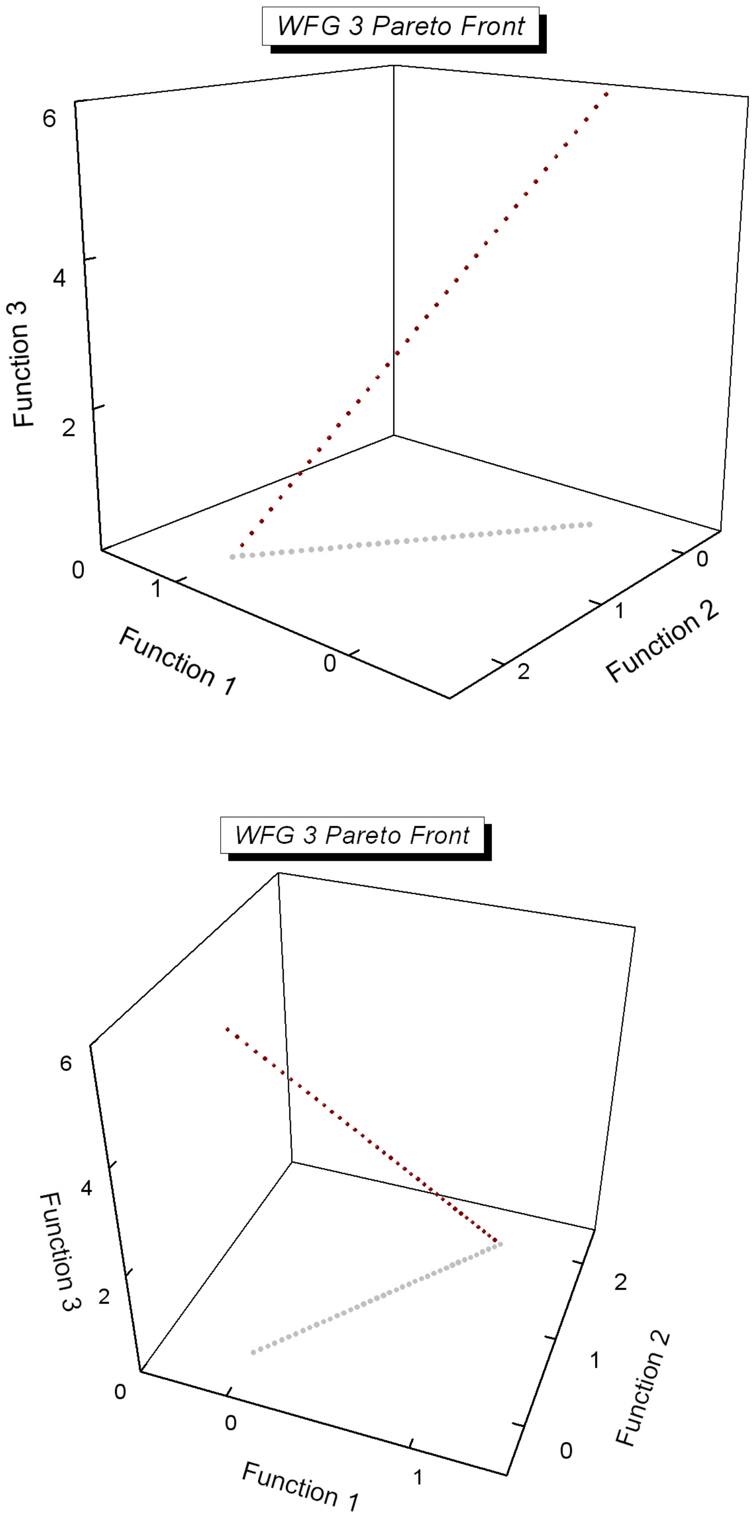


Figure 2.14: 3-objective WFG3 Pareto Front

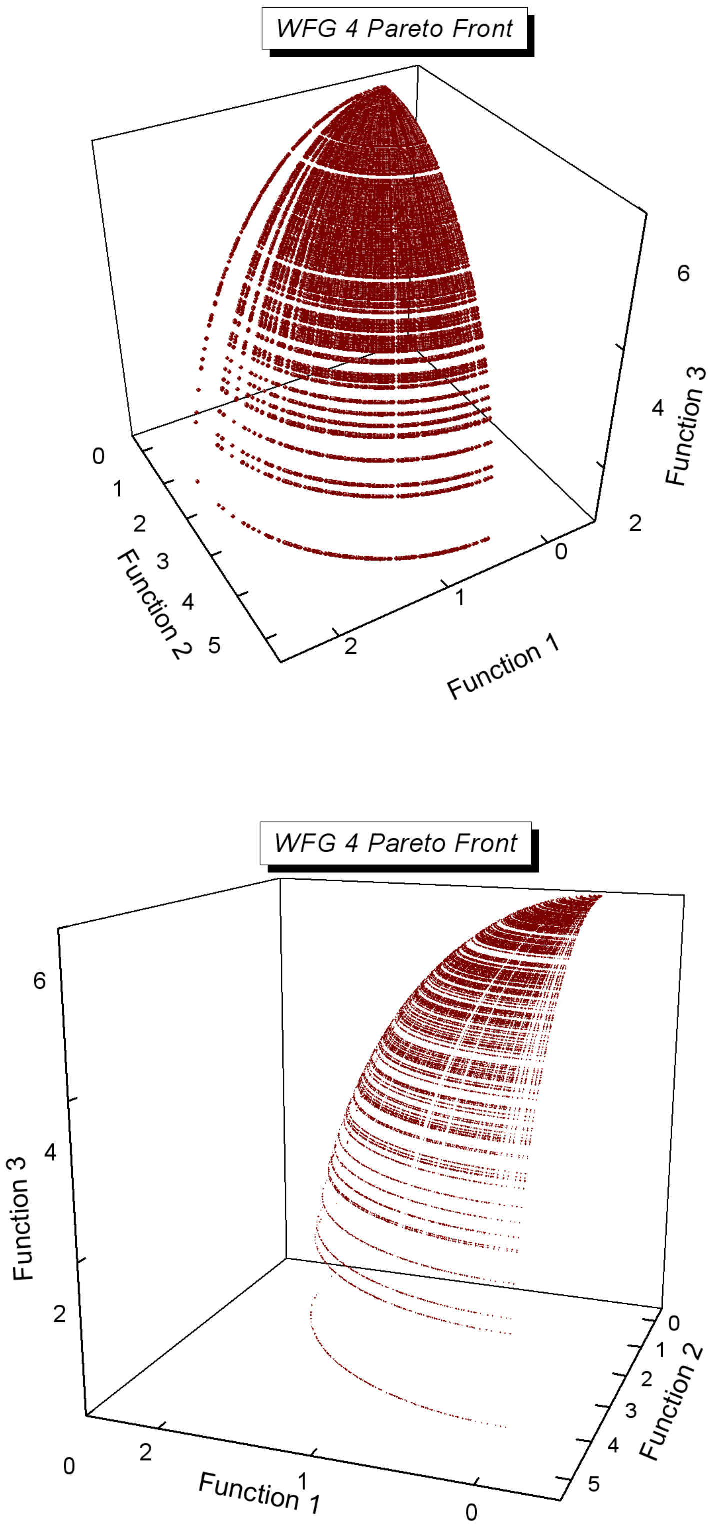


Figure 2.15: 3-objective WFG4 Pareto Front

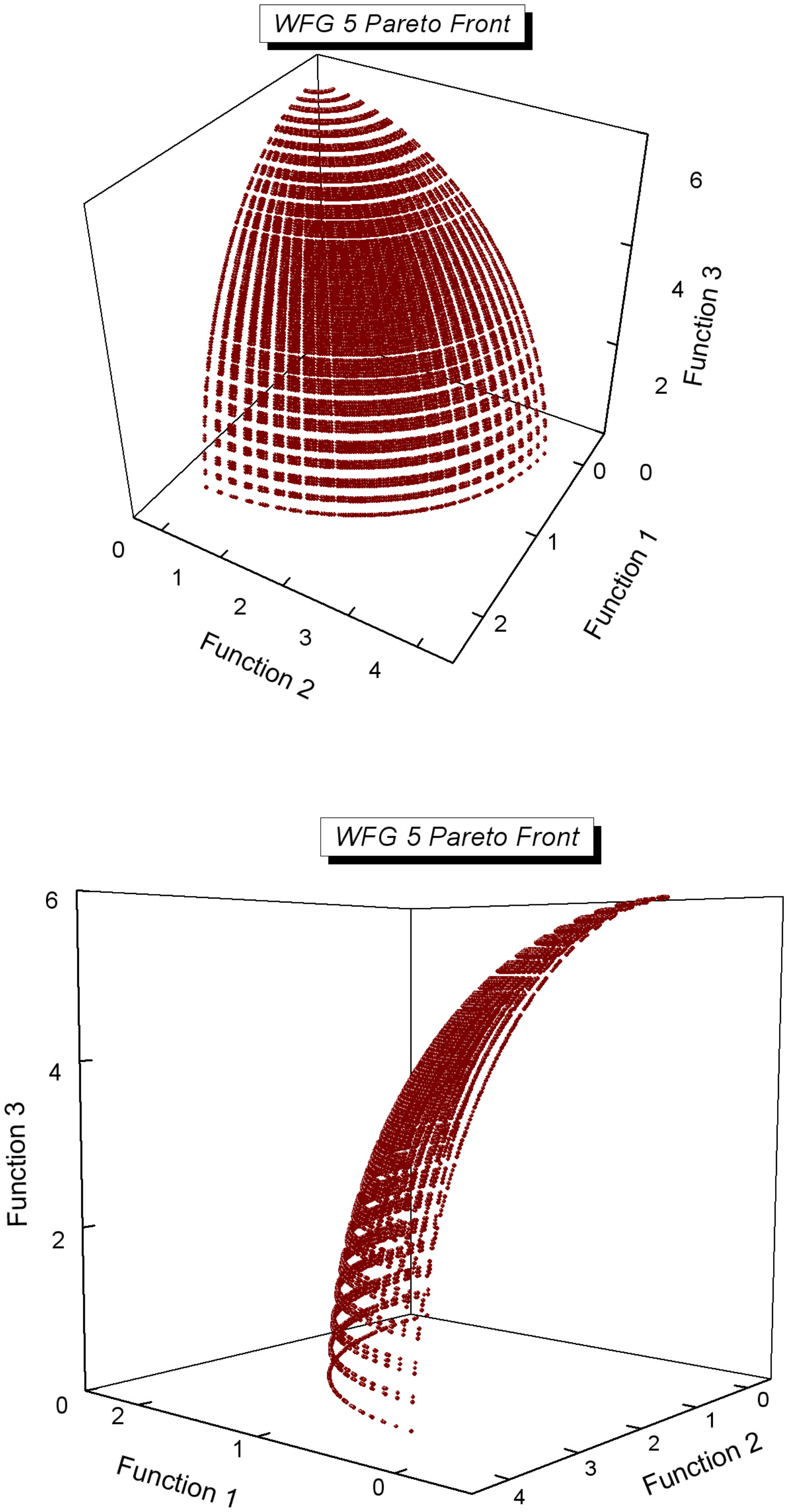


Figure 2.16: 3-objective WFG5 Pareto Front

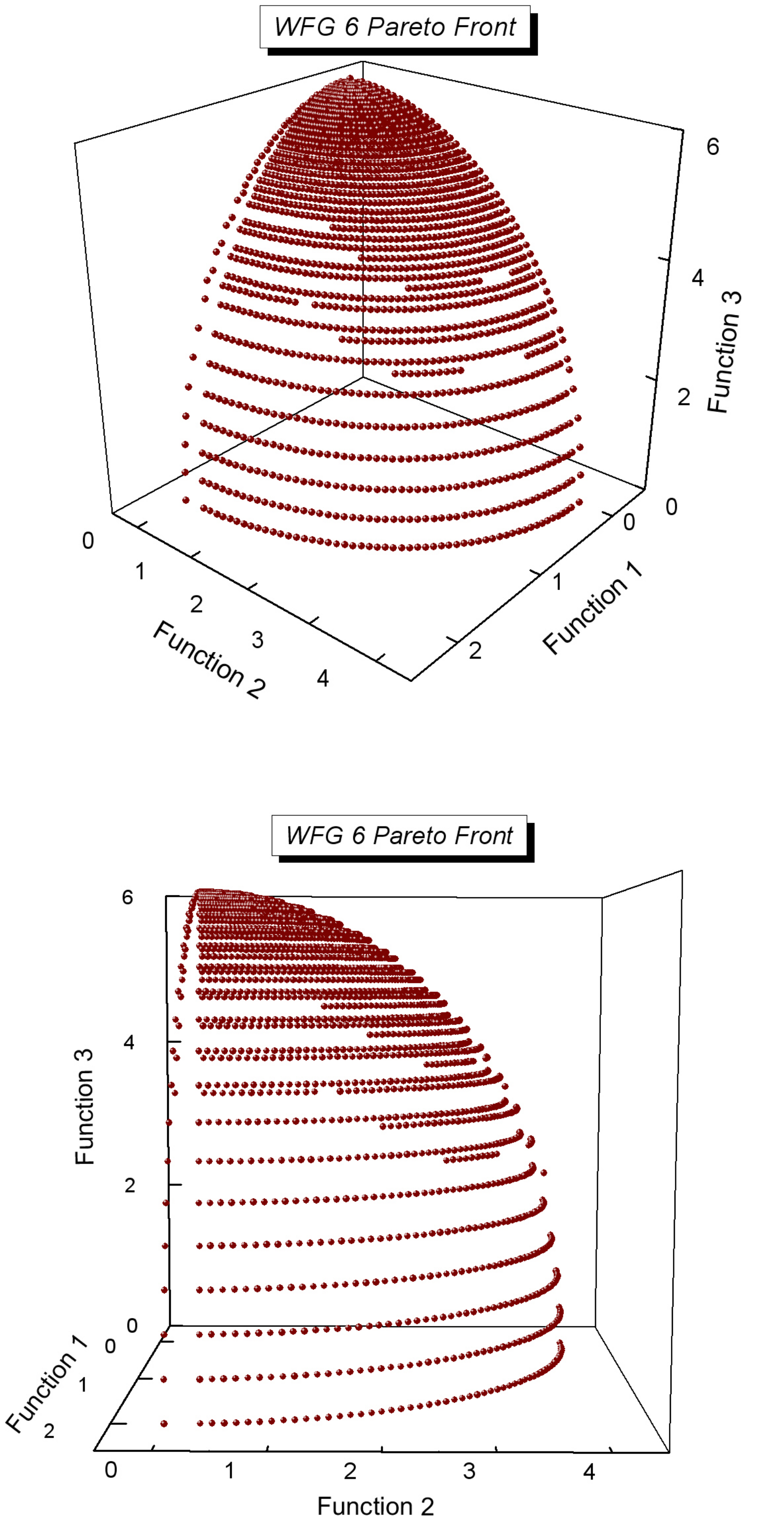


Figure 2.17: 3-objective WFG6 Pareto Front

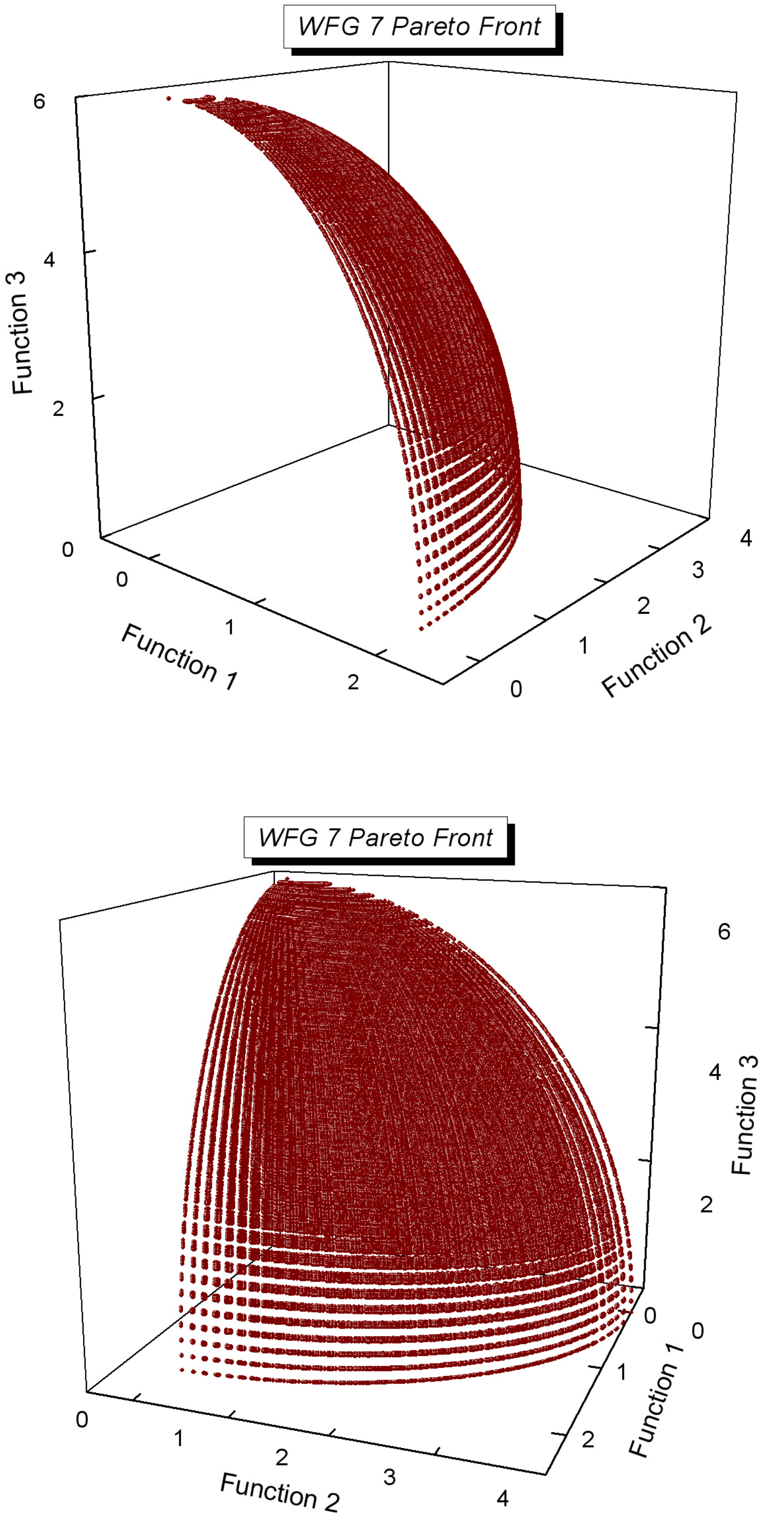


Figure 2.18: 3-objective WFG7 Pareto Front

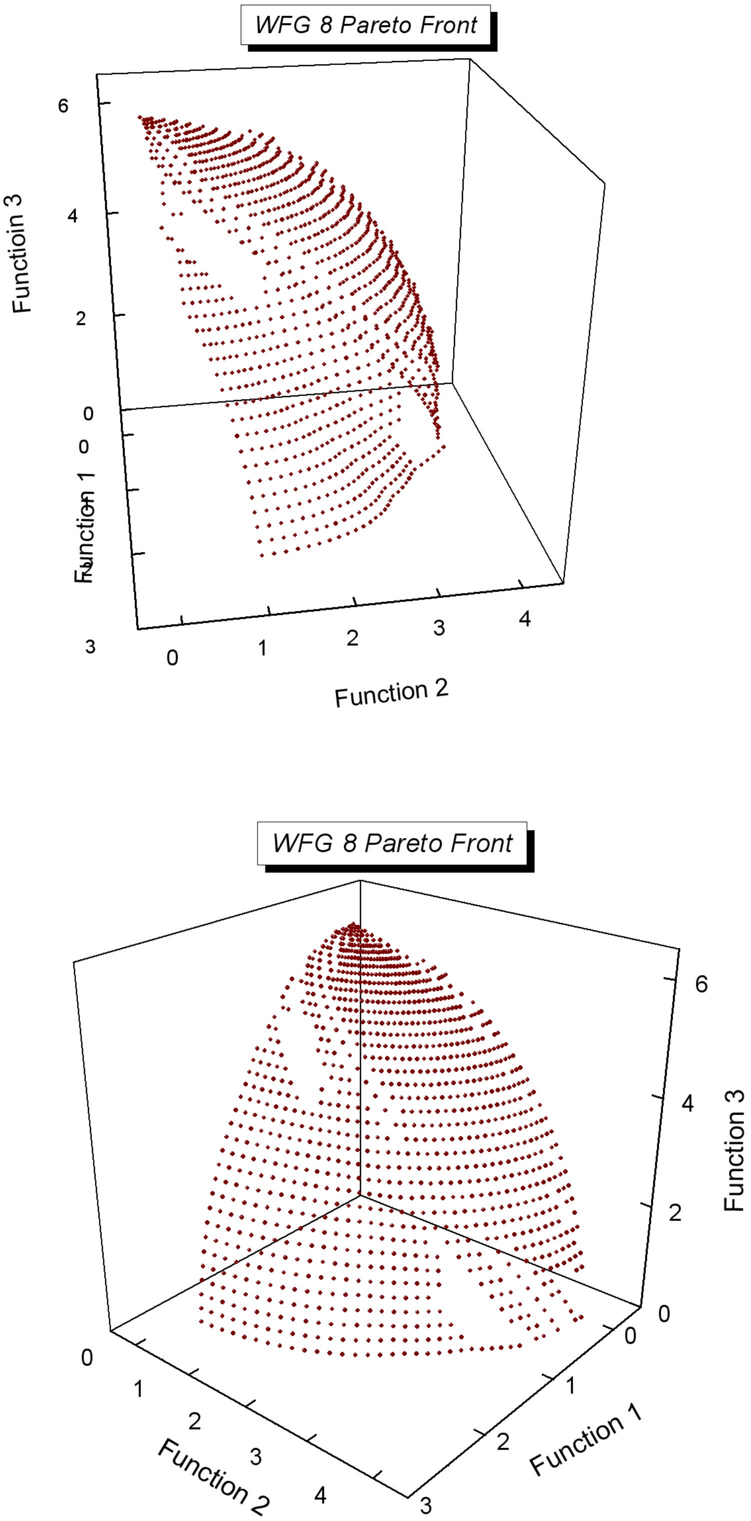


Figure 2.19: 3-objective WFG8 Pareto Front

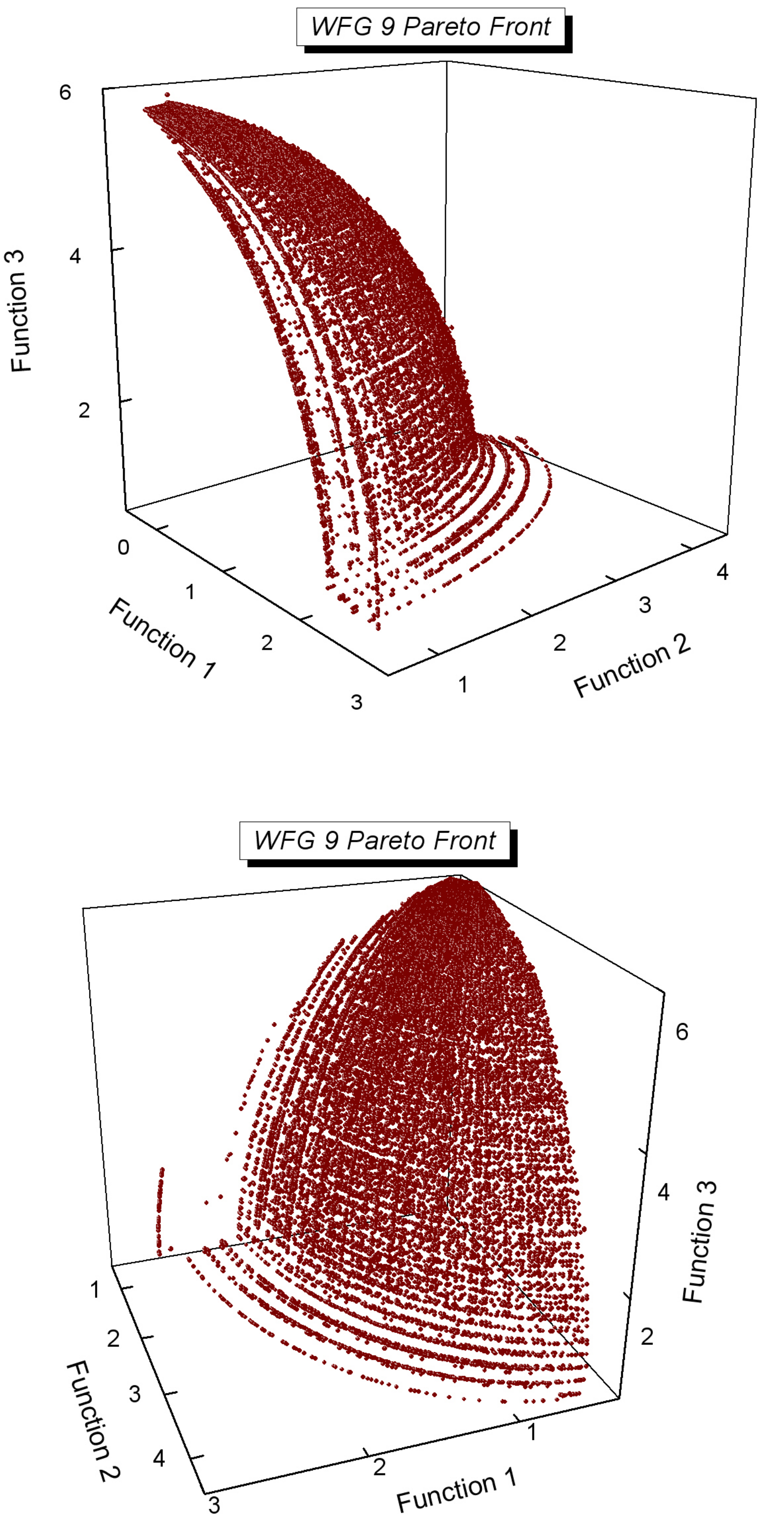


Figure 2.20: 3-objective WFG9 Pareto Front

2.4 Performance Measures

Empirical comparison of different optimization algorithms require some way to quantify the performance quality of an algorithm. In the case of MOO and MaOO, quantification of performance quality is substantially more complex than for a UOO. This is due to the multiple, often contradictory, objectives present in MOPs and MaOPs. Remember from section 2.1.3 that an ideal POF has a good spread of solutions that are as close as possible to the true POF. There exist several quality indicators that either measure closeness to the true POF (such as generational distance [160, 161]) or diversity among the found POF (such as maximum spread [151, 177], spacing [138], solution distribution [63], and spread [35]). However, this study chose two performance measures that each takes into account both solution accuracy and diversity. The chosen performance indicators are discussed next in section 2.4.1 and section 2.4.2.

2.4.1 Hypervolume

The HV measure was introduced by Zitzler and Thiele [180] with the purpose of quantifying the objective space volume covered by the hypercubes, $\mathcal{V}_k$, for each solution $\mathbf{f}(\mathbf{x})_k \in Q$, where $\mathcal{V}_k$ is the hypercube constructed between the reference-point $\mathbf{r}_{hv}$ (originally defined as the zero vector, $\mathbf{0}$) and the solution $\mathbf{f}(\mathbf{x})_k$. Due to overlapping hypercube volumes, the HV measure calculates the volume of the union of all the hypercubes in the objective space [180]. Formally, the HV measure is calculated as

$$HV = \text{volume}(\bigcup \mathcal{V}_k), \forall\, \mathbf{f}(\mathbf{x})_k \in Q \tag{2.38}$$

Van Veldhuizen [159] observed that the HV measure can be spurious if the POF is non-convex. That is, when using the zero vector as the reference-point, the HV would decrease as the quality of the POF improves, which is misleading since a larger HV should be indicative of a better POF. Zitzler [177] concluded that Van Veldhuizen's [159] finding indicates that the coverage of the objective space is one of several viable criteria to be considered during the evaluation of the POF. An alternative approach, improving upon that of Ziztler and Thiele [180], uses the nadir vector, $\mathbf{z}^{\text{nad}}$, as the reference-point. When using the nadir vector as the reference-point, the HV increases as the quality of the POF improves. More recently [106], and in this study, the reference-point is chosen to be a vector with values slightly larger than the normalized nadir point, i.e., $\mathbf{1.1}$. This vector has been shown to appropriately accentuate the convergence and diversity of the found solutions [3] by including the HV contribution of each extreme Pareto-optimal solution [79]. The computational cost of the HV calculation increases as the number of found solutions and objectives increase. The HV measure can, however, be estimated when the number of objectives is more than some threshold. The HV calculation can be approximated using the Monte Carlo sampling technique [4]. Bader and Zitzler recommend at least 10 000 sampling points [4] for the Monte Carlo approach. Note that a more computational friendly, recursive, dimension-sweep HV algorithm [58] exists. However, this algorithm [58] is still extremely time-consuming for MaOPs with a large number of objectives such as 10 or 15. Also, note that other faster calculation algorithms do exist [9].

2.4.2 Inverted Generational Distance

The IGD measure was introduced by Coello Coello and Reyes-Sierra [25, 128]. The IGD measure is calculated as

$$IGD = \frac{\sqrt{\sum_{k=1}^{|POF^*|} b_k^2}}{|POF^*|} \tag{2.39}$$

where POF^* is a set of Pareto-optimal solutions representing the true POF, and b_k is the Euclidean distance in the objective space between solution k of the true POF and the nearest solution of the known front, $\mathcal{Q}$. Note that $\mathcal{Q} = POF^*$ when $IGD = 0$.

The quality of the IGD measure depends on the quality of the known POF. Ishibuchi *et al.* [80] investigated the challenges in specifying the reference-points making up the true POF. Ishibuchi *et al.* [80] found that uniform sampling of reference-points from the known true POF leads to counter-intuitive results. For all IGD calculations, this work made use of the true POFs generated by the PlatEMO framework [153] – an evolutionary multi-objective optimization platform.

2.5 Summary

The goal of this chapter was to provide background information about optimization in general, multi-objective optimization (MOO), multi-objective optimization problems (MOPs), many-objective optimization (MaOO), many-objective optimization problems (MaOPs), test problems, and MaOO performance measures. More specifically, a MOP was defined formally. Next, approaches for defining optimality for optimization problems with more than one objective were defined and discussed, which included the weighted aggregation and Pareto-optimality approach. The formal definition of a MaOP was also presented followed by the challenges faced by MaOO algorithms when solving MaOPs. Different approaches to MaOO were also discussed. Finally, the benchmark problems, as well as the performance measures used in this study to help evaluate the investigated algorithms, were formally defined and discussed.

The next chapter provides further background information related to the MaOO algorithms that formed part of this investigation.

Chapter 3

Many-objective Optimization Algorithms

"Truly successful people have learned to do what does not come naturally. Real success lies in experiencing fear or aversion and acting in spite of it."

— Dr Joseph Mancusi

An optimization algorithm is a search method with the purpose of finding an optimal solution by the process of iteratively transforming a current candidate solution with the hope of improving the quality of that candidate solution. The end goal for MOO and MaOO algorithms is to find a set of solutions which cannot improve any objective without degrading some of the other objectives. When considering all possible solutions, a set like this is deemed Pareto-optimal. This study investigates stochastic, local search, optimization algorithms. Stochastic algorithms incorporate random elements when modifying candidate solutions. Local search algorithms only use locally available search space information by exploiting neighbourhoods; this implies that only local optima discovery can be guaranteed. In contrast, deterministic algorithms do not use random elements and global search algorithms can guarantee global optimal solutions by exploring the entire search space. Furthermore, each of the algorithms investigated in this study can be classified as either an evolutionary algorithm (EA) or a particle swarm optimization (PSO) algorithm. To better understand where these algorithms fit into the vast field of artificial intelligence (AI), the reader is referred to [47], where a classification of computational intelligence (CI), a sub-branch of AI, paradigms are sufficiently covered. There are numerous MaOO algorithms, however, this study selected some of the well-performing competitive MaOO algorithms; that is, MOO algorithms that have been shown to scale well to MaOPs. The interested reader is referred to [107] where over a hundred MaOO algorithms are categorized. Note that this list of MaOO algorithms [107] is non-exhaustive and ever-growing.

Section 3.1 introduces the basic EA followed by a discussion highlighting the most prominent features of each of the many-objective evolutionary algorithms (MaOEAs) investigated in this study. Section 3.2 introduces the basic PSO algorithm, followed by a discussion highlighting the most prominent features of the only many-objective particle swarm optimization (MaOPSO) algorithm investigated in this study. The focal algorithm of this thesis, the MGPSO algorithm (an aspiring MaOPSO algorithm), is also discussed in section 3.2. Finally, this chapter is epitomized in section 3.3.

3.1 Evolutionary Algorithms

This section focusses on EAs, which are part of the evolutionary computation (EC) paradigm, a sub-branch of CI [47]. EC aims to model the concept of natural evolution. EC has been used successfully in real-world applications, for example, data mining, combinatorial optimization, fault diagnosis, classification, clustering, scheduling, and time series approximation [47]. Section 3.1.1 introduces the original EA and section 3.1.2 gives a high-level overview of the MaOO EAs relevant to this study.

3.1.1 Basic Evolutionary Algorithm

An evolutionary algorithm (EA) aims to imitate biological evolution as defined by the Darwinian theory of natural selection [28]. Natural selection is established on the concept of survival of the fittest where the weak perish. EAs use a randomly chosen population of chromosomes, where each chromosome represents an individual. An individual is simply a candidate solution to the problem at hand. A chromosome consists of genes, where each gene describes a characteristic of that individual in the population. The value of a gene is also known as an allele. For each generation (or algorithm iteration), the individuals of a population compete for the privilege of being allowed to reproduce offspring. The fittest individuals of the population have a higher probability of being selected to reproduce offspring. Crossover is the process of creating new individuals (offspring) by combining the desirable genes from two or more parents. In an effort to promote diversity, each individual of the population can also undergo mutation which alters some of the alleles of the chromosome. The fitness of an individual is measured using a function which takes into account both the objectives and constraints of the problem. At the end of each generation, a new population can be selected from the offspring or, from both the parents and the offspring - as long as the fittest individuals survive to the next generation(s). Therefore, an EA is a stochastic search algorithm that searches through the search space of viable chromosome values for an optimal combination of genes. The pseudocode for a generic EA is presented in Algorithm 1.

Algorithm 1 Evolutionary Algorithm (EA)

1 : Let $t = 0$ be the generation counter;
2 : Create and initialize an n-dimensional population, $\mathcal{C}(0)$, to consist of n_s individuals;
3 : **while** stopping condition(s) not true **do**
4 : Evaluate the fitness, $f(\mathbf{x}_i(t))$, of each individual, $\mathbf{x}_i(t)$;
5 : Select parents to perform reproduction (i.e. crossover) at a probability to create offspring;
6 : Apply mutation to the population at some probability;
7 : Select the new population, $\mathcal{C}(t+1)$;
8 : Advance to the new generation, i.e. $t = t + 1$;
9 : **end while**
10 : **return** $\mathcal{C}$;

Note that at line 6 of Algorithm 1 a low probability is often preferred. However, occasionally, an EA may be more effective if a large mutation probability is used initially (to promote exploration) which is then gradually decreased over time as the search progresses (to promote exploitation). In other words, the mutation probability, to some degree, is problem-dependent.

The following section gives an overview of the benchmark MaOEAs used in this study.

3.1.2 Many-objective Evolutionary Algorithms

Since the first EA [12], innovation and research of more complex problems have led to the development of MOEAs as well as MaOEAs. MaOEAs have the ability to solve problems with two or more objectives; such as MOPs and MaOPs. MOEAs, on the other hand, can only solve MOPs effectively. Sections 3.1.2.1 to 3.1.2.3 give an overview of the three benchmark MaOEAs used in this work, focussing on the most important and unique features that help each algorithm to scale. All MaOEAs considered in this study are at their core elitist, i.e., the average fitness value of the population will never decrease as the search progresses. Elitist EAs ensures that a certain number of the fittest individuals are chosen from the union of the current population set and the offspring set to survive to the next generation.

3.1.2.1 Knee-point driven Evolutionary Algorithm

The knee-point driven evolutionary algorithm (KnEA) [174] primarily uses the Pareto-dominance relationship during selection. However, the KnEA also utilizes a secondary selection (convergence) criterion, referred to as knee-points, that assists the Pareto-dominance relationship which is known to degrade as the number of objectives increases [68, 77, 81, 106, 141, 174].

Knee-points are a subset of the Pareto-optimal solutions for which an improvement in one objective will result in an acute deterioration in at least one other objective [174]. The KnEA incorporates knee-points with the purpose of increasing its selection pressure; i.e., the ability of an algorithm to converge to the true POF.

The importance of knee-points has long been recognized and has, therefore, lead to the development of several knee-point identification methods [8, 11, 29, 37, 40, 105, 124, 125, 142, 174]. Some problems have also been developed to test the ability of algorithms to find knee-points [11, 155]. Note, however, that these problems are not scalable, i.e., these problems are limited to two, three, four, and five objectives.

The KnEA uses an adaptive strategy for identifying knee-points as defined by Das [29] and Bechikh *et al.* [8]. To identify a knee-point, this approach first determines a set, E, of extremal solutions, where an extremal solution E_m is defined as the solution with the least desirable fitness for objective m. Next, a hyperplane H is created as follows:

1. Create all possible $n_m - 1$ distinct vectors using the solutions of E.
2. Construct a matrix $\mathbf{G}$, where each row of $\mathbf{G}$ is a vector computed in the previous step.
3. Compute the null space of $\mathbf{G}$, using the result to determine the constants of the hyperplane H.

H can then be used to determine the knee-points of the approximated (found) front using the following method: for each solution $\mathbf{x}$, $\mathbf{x}$ is considered a knee-point if, and only if, $\mathbf{x}$ possesses the maximum objective space distance to H within its neighbourhood. An adaptive strategy is used to determine the neighbourhood size of each solution. The neighbourhood of a solution is a hypercube with n_m sides, where each side is calculated as

$$R_m(t) = \big(max_m(t) - min_m(t)\big) \cdot ratio(t) \tag{3.1}$$

where $max_m(t)$ and $min_m(t)$ denote the maximal and minimal values of the m-th objective at iteration t respectively and $ratio(t)$ corresponds to the ratio of the neighbourhood size to the range spanned by objective m at iteration t. $ratio(t)$ is updated using

$$ratio(t) = ratio(t-1) \cdot e^{-\frac{1-(\zeta(t-1)/\kappa)}{nm}} \tag{3.2}$$

where $\zeta(t-1)$ denotes the ratio of knee-points to non-dominated solutions at iteration $t-1$ and κ is a user-defined parameter which represents the desired ratio of knee-points to non-dominated solutions, with $0 < \kappa < 1$. Note that initially $\zeta(t) = 0$ and $ratio(t) = 1$. The strategy above essentially shrinks and grows the neighbourhood size adaptively until the ratio of knee-points to non-dominated solutions in the solution set converges to κ.

Importantly, it is made clear in [174] that the KnEA identifies the knee-points within the found POF and not the true (theoretical) POF. An illustration of determining knee-points for a bi-objective minimization problem is illustrated in figure 3.1. In this example, solutions B, E, and G are identified as knee-points because each of these solutions possesses the largest objective space distance to the extreme hyperplane H within their neighbourhood. Note that if a solution is isolated within its neighbourhood, e.g., solution G, this solution is also considered to be a knee-point. The above knee-point definition leads to the benefit that the diversity of the population is implicitly taken into account. Note that extreme solutions are also considered knee-points, as depicted in figure 3.1.

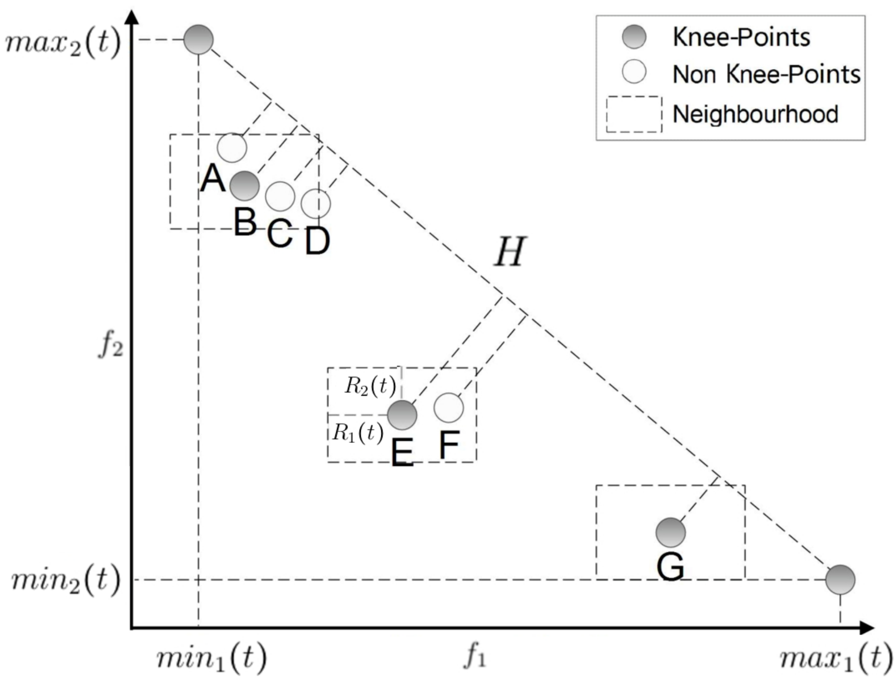


Figure 3.1: Illustration for determining knee-points.

The main components of the KnEA are presented in Algorithm 2. The KnEA starts by first setting the generation counter and then initializes the population by generating a random set of solutions (lines 1 and 2).

Second, simulated binary crossover (SBX) [2] and polynomial mutation [36] are performed to generate the offspring individuals (lines 4 and 5). Mutation is used to introduce variation into the population which enhances solution diversity and promotes exploration. Mutation is usually applied at a low probability as to not completely transform already promising solutions. During crossover (line 4), three tournament metrics are adopted, namely, the Pareto-dominance relationship, the knee-point criterion, and the weighted distance measure. The weighted distance measure, like crowding distance, is an alternative way of measuring the crowdedness of a solution. The weighted distance measure is based on the k-nearest neighbours algorithm [56]. The reader is referred to [174], where the details of the weighted distance metric are discussed. During the binary tournament mating selection, the KnEA randomly selects two individuals from the parent population. If one solution dominates the other solution, then the former solution is chosen as the winner. When performing a binary tournament between two mutually non-dominated solutions, if either solution is a knee-point it is regarded as more desirable and wins the tournament. If both of the solutions are knee-points or neither is a knee-point, then a weighted distance is used for comparing the two solutions. The solution with the larger weighted distance wins the tournament. If both solutions have an equal weighted distance, then one of the solutions is randomly chosen for reproduction.

Third, non-dominated sorting is performed which sorts the combined parent and offspring population into several non-domination levels (line 6). Solutions in the first level have the highest priority to be selected as the next parents. Next an adaptive strategy is used to identify solutions located in the knee-regions of each non-dominated front in the combined population (line 7). The KnEA does not require an additional diversity mechanism (such as crowding distance) since the knee-point identification strategy used implicitly takes diversity into account by ensuring that each neighbourhood has exactly one knee-point [174]. Also, this knee-point identification strategy used by the KnEA can identify false knee-points, which further promotes diversity through exploration, and can speed up convergence towards the POF [174].

Fourth, environmental selection is conducted to select the fittest individuals as the parent population of the next generation (lines 8 and 9). Note that knee-points are also employed during the environmental selection phase of the KnEA. During the environmental selection phase, the KnEA starts to select the non-dominated solutions in the first non-dominated front. If the number of solutions in the first non-dominated front is larger than the population size, which is very likely already in the early generations in MaOO, then knee-points in the first non-dominated front are selected first as parents for the next population. If the number of knee-points in the first non-dominated front is larger than the population size, then n_s knee points having a larger distance to the extremal hyperplane are selected. Otherwise, the knee points are selected together with the remaining solutions in the first non-dominated front that have a larger distance to the hyperplane of the first non-dominated front. If the number of solutions in the first non-dominated front is smaller than n_s, the KnEA turns to the second non-dominated front for selecting the remaining parent solutions. If the second non-dominated front is larger than the number of remaining population slots, then the same procedure described above will be applied to the second non-dominated front. This process is repeated until the parent population for the next generation reaches n_s.

This procedure repeats until one or more stopping conditions are met. Finally, the KnEA terminates after returning the final population, i.e., the solutions representing the found POF (line 11).

Algorithm 2 Knee-point driven Evolutionary Algorithm (KnEA)

1 : Let $t = 0$ be the generation counter;
2 : Create and initialize an n-dimensional population, $\mathcal{C}(0)$, to consist of n_s individuals;
3 : **while** stopping condition(s) not true **do**
4 :   Mating selection;
5 :   Generate offspring using a variation method;
6 :   Perform non-dominated sorting;
7 :   Identify knee-points;
8 :   Perform environmental selection to select the new population, $\mathcal{C}(t+1)$;
9 :   Advance to the new generation, i.e. $t = t + 1$;
10 : **end while**
11 : **return** $\mathcal{C}$;

It is shown in [174] that favouring knee-points in the found POF results in an approximation of a bias towards a larger HV. This is a desirable property since the HV calculation cannot guarantee diversity even though it implicitly takes diversity into account [174]. The KnEA has also been shown to be very competitive both in terms of solution quality and computational efficiency when solving MaOPs [106, 174]. Therefore, the KnEA was selected as one of the benchmark algorithms in this study. Also, note that this same adaptive knee-points identification approach is used similarly in this study to investigate if it helps with the scalability of the MGPSO algorithm. For more detail about the KnEA, the reader is referred to [174].

3.1.2.2 Many-objective Evolutionary Algorithm based on Dominance and Decomposition

Decomposition refers to decomposing the MOP or MaOP being solved into a set of scalar optimization subproblems which is then optimized cooperatively, allowing for better convergence [95]. The many-objective evolutionary algorithm based on dominance and decomposition (MOEA/DD) [95] combines the strengths of the well-known non-dominated sorting genetic algorithm (NSGA-II) [35] and the multi-objective evolutionary algorithm based on decomposition (MOEA/D) [173], which have been shown to excel at different problems [171]. The MOEA/DD exploits the desirable properties of the Pareto-dominance relationship, decomposition, and local density estimation to balance convergence and diversity during the evolutionary search process [95]. Local density estimation is used to determine the density (crowdedness) of solutions within a subregion of the search space.

The MOEA/DD uses the promising penalty-based boundary intersection (PBI) decomposition method [95, 173]. The PBI aggregation method is defined as

$$\text{minimize} \quad g^{pbi}(\mathbf{x}|\mathbf{w},\mathbf{z}^*) \quad = \quad d_1 + \varphi d_2 \tag{3.3}$$

where

$$\begin{aligned} d_1 &= \frac{\|(\mathbf{f}(\mathbf{x})-\mathbf{z}^*)^{\psi}\mathbf{w}\|}{\|\mathbf{w}\|} \\ d_2 &= \left\|\mathbf{f}(\mathbf{x}) - \left(\mathbf{z}^* + d_1\frac{\mathbf{w}}{\|\mathbf{w}\|}\right)\right\| \end{aligned}$$

and $\varphi \geq 0$ is a user-defined penalty parameter that controls the balance between d_1 (i.e. the convergence component) and d_2 (i.e. the diversity component); ψ is the neighbourhood size; $\mathbf{w}$ is a weight vector associated with solution $\mathbf{x}$. By modifying the choice of $\mathbf{w}$, different Pareto-optimal points (solutions) can be found.

Algorithm 3 presents the pseudocode for the MOEA/DD. First, the initialization procedure (lines 2 and 3) generates the initial population, $\mathcal{C}(0)$, and a set of weight vectors, W. The weight vectors are generated to be widely spread using the boundary intersection method of Das and Dennis [31]. The details of this weight generation approach used by the MOEA/DD is discussed in detail in [95] and [31]. Each weight vector defines a subproblem that can be used for fitness evaluation and also represents a unique subregion of the objective space that can be used to estimate the local population density [95]. A subregion consists of a number of the closest weight vectors evaluated by Euclidean distances. Next, the non-domination level structure of the population is determined using the fast non-dominated sorting method [35] (line 4). Finally, each individual of the population is randomly assigned to (or associated with) a unique subregion (line 5).

Within the while-loop, in case that one or more termination criteria are not met, for each weight vector, the mating selection procedure chooses neighbouring individuals as parents for offspring generation (line 7). Neighbouring solutions are located in neighbouring subregions. This allows for more efficient mating in a high-dimensional search space; that is, producing offspring close to the parents in the objective space [95]. In case no associated solution exists in the selected subregions, mating parents are randomly chosen from the whole population. Moreover, to enhance exploration [94], mating between parents selected from the whole population is allowed at a low probability.

Next, offspring is generated by performing SBX [2] and polynomial mutation [36] (line 8). Mutation is applied to the individuals (usually at a low probability) for diversity and exploration purposes.

After sorting the population into non-dominated levels (line 9), each individual in the population is associated with a weight vector (subregion) based on Euclidean distance (line 10). To help distinguish solutions, the MOEA/DD performs local density estimation by incorporating a local niche count (line 11) for each subregion (i.e. the number of solutions associated with a subregion) [95]. Local density estimation promotes diversity by allowing poor solutions to survive to the next generation if the solution is located in an isolated subregion; that is, if the solution is located in a subregion with a low niche count [95]. Then the PBI value for each individual is calculated (line 12).

Next, the parent population is updated using a steady-state scheme (line 13), where only one offspring solution is considered each time [95]. In the case of only one non-dominated front, the solution located in the most crowded subregion, i.e., the neighbourhood with the largest niche count is chosen as the worst solution. In any case, where more than one subregion has the same largest niche count, the MOEA/DD chooses the solution with the largest PBI value in the subregion with the largest sum of PBI values as the worst solution. In the case that more than one non-domination level (front) exists, culling of solutions start at the last (least desirable) non-domination level. A solution is culled from the population if it is the only solution in the last non-domination level that is located in a subregion containing other better converging solutions (i.e. solutions in better non-domination levels). If the only solution in the worst non-domination level is located in an isolated subregion it is definitely preserved and the solution in the next best (second last) non-domination level with the largest PBI value in the most crowded subregion is removed. In the case of more than one non-domination levels and more than one solution in the last (worst) non-dominated level, the solution with the largest PBI value in the most crowded subregion is chosen to be the worst solution - the solution to be eliminated. If the niche count of the most crowded subregion is one, it means that every member in the current worst non-domination level is associated with an isolated subregion. As discussed before, such solutions should be preserved for the next round without reservation and the solution in the worst non-domination level with the largest PBI value in the most crowded subregion is rather removed.

This procedure repeats until one or more stopping conditions are met, at which point the MOEA/DD algorithm will return the final population, representing the found POF (line 16).

Algorithm 3 Many-objective Evolutionary Algorithm based on Dominance and Decomposition (MOEA/DD)

1 : Let $t = 0$ be the generation counter;
2 : Create and initialize an n-dimensional population, $\mathcal{C}(0)$, to consist of n_s individuals;
3 : Generate an m-dimensional set of weight vectors, W, to consist of n_s weight vectors;
4 : Perform non-dominated sorting on the population;
5 : Randomly associate each individual with a unique neighbourhood;
6 : **while** stopping condition(s) not true **do**
7 :   Mating selection;
8 :   Generate offspring using a variation method;
9 :   Perform non-dominated sorting;
10 :   Associate each individual with a neighbourhood;
11 :   Compute niche counts;
12 :   Compute PBI values;
13 :   Select the new population, $\mathcal{C}(t+1)$;
14 :   Advance to the new generation, i.e. $t = t + 1$;
15 : **end while**
16 : **return** $\mathcal{C}$;

The MOEA/DD has been shown to work well for MaOPs [95, 106]. Therefore, the MOEA/DD was chosen to be part of the set of benchmark algorithms used in this study. The reader is referred to [95] for greater detail about the different components of the MOEA/DD.

3.1.2.3 Reference-point based Many-objective Non-dominated Sorting Genetic Algorithm

A genetic algorithm (GA) [71] is a specific type of EA. The reference-point based many-objective non-dominated sorting genetic algorithm

(NSGA-III) [38] favours individuals of a population which are non-dominated and also near a set of supplied well-spread reference-points [38].

Algorithm 4 presents the pseudocode for the NSGA-III. The NSGA-III starts with a similar initialization process as the MOEA/DD (lines 1 through 5). The NSGA-III uses the structured reference-point creation procedure proposed by Das and Dennis [31]. The details of this weight generation approach used by the NSGA-III (and the MOEA/DD) is discussed in detail in [38] and [31]. Note that, with some domain knowledge, the reference-points can also be supplied by the user [38].

The NSGA-III uses the same crossover and mutation operators as the KnEA and the MOEA/DD, i.e., SBX [2] and polynomial mutation [36] (lines 7 and 8). Next, the NSGA-III ranks the population by performing a non-dominated sort which divides the population into non-domination levels (line 9).

Furthermore, for each generation, the NGSA-III normalizes each objective function value using the extreme solutions found since the start of search (line 10). This allows the NSGA-III to solve MaOPs whose objective values may be differently scaled [38].

Next, each individual is associated with one of the reference-points (line 11). An individual is associated with the closest reference-point in terms of Euclidean distance in the normalized objective space. It is worth noting that a reference-point may have one or more population members associated with it or need not have any population member associated with it. Then the niche count for each reference-point (neighbourhood or subregion) is calculated (line 12).

A niche-preserving operation is performed to select the fittest diverse individuals for the next generation (line 13). This is done by counting the number of individuals associated with each reference-point. In essence, this process is used to preserve individuals that are associated with a reference-point with a low niche count. The details of the niche-preservation operation can be viewed in [38]. This approach not only speeds up the search process (convergence to the POF) but also ensures diversity if the specified reference-front is diverse [38]. Therefore, if a diverse set of reference-points is used there is no need for other computationally expensive diversity management approaches such as, for example, the crowding distance operator [38].

This process is repeated until one or more stopping conditions are reached, at which point the algorithm returns the approximated POF (line 16).

Note that the NSGA-III and the MOEA/DD have some similarities:

- Both employ a set of weight vectors (reference-points) to guide the selection procedure.
- Both algorithms associate each solution with a weight vector (reference-point).

Algorithm 4 Reference-point based many-objective Non-dominated Sorting Genetic Algorithm (NSGA-III)

1 : Let $t = 0$ be the generation counter;
2 : Create and initialize an n-dimensional population, $\mathcal{C}(0)$, to consist of n_s individuals;
3 : Generate an m-dimensional set of reference-points, W, to consist of n_s weight vectors;
4 : Perform non-dominated sorting on the population;
5 : Randomly associate each individual with a unique neighbourhood;
6 : **while** stopping condition(s) not true **do**
7 :   Mating selection;
8 :   Generate offspring using a variation method;
9 :   Perform non-dominated sorting;
10 :   Perform adaptive normalization;
11 :   Associate each individual with a reference-point;
12 :   Compute niche counts;
13 :   Select the new population, $\mathcal{C}(t+1)$;
14 :   Advance to the new generation, i.e. $t = t + 1$;
15 : **end while**
16 : **return** $\mathcal{C}$;

- Both divide the population into several non-domination levels according to the Pareto-dominance relation. Note that this is also the case for the KnEA.

The NSGA-III and the MOEA/DD also have subtle differences which include:

- With the MOEA/DD a solution associated with an isolated subregion will survive to the next iteration even if it belongs to the last non-domination level. In contrast, the NSGA-III does not allow such a solution to survive to the next generation even if it could promote population diversity.

- In the MOEA/DD, each weight vector not only specifies a unique subregion in the objective space but also defines a subproblem which can be used to evaluate the fitness value of a solution (Equation 3.3).

The NSGA-III has been shown to perform well on MaOPs [38, 106] and, therefore, was also selected as a benchmark algorithm for this work. The reader is referred to [38] for greater detail about the NSGA-III.

3.2 Particle Swarm Optimization

This section focusses on PSO algorithms, which are part of the swarm intelligence (SI) paradigm, a sub-branch of CI [47]. SI is rooted in the study of colonies, or swarms of social organisms. PSO has been successfully applied to a variety of real-world problems, for example, neural network training, optimizing equipment design parameters, optimizing space mission fuel expenditure models, clustering, design, scheduling, planning, controllers, power systems, bioinformatics, and data mining [47]. PSO has also been shown to outperform traditional EC algorithms on complex problems [87]. PSO can also, in principle, be applied to any problem that can be expressed in terms of an objective function to be optimized. PSO does not use gradient information, making it suitable for problems where no gradient information is available or where the objective function is not differentiable. In other words, PSO can be used to solve black-box optimization problems. Note that PSO is discussed in greater detail than EAs, because PSO is foundational to the central algorithm of this work, i.e., the MGPSO algorithm.

Section 3.2.1 introduces the original PSO algorithm and related concepts. Section 3.2.2 gives a high-level overview of the benchmark MaOPSO algorithm used in this study as well as the focal, aspiring MaOPSO algorithm, the MGPSO algorithm.

3.2.1 Basic Particle Swarm Optimization

Kennedy and Eberhart [86] introduced PSO in 1995. PSO is rooted in studies that simulated bird flocking behaviour when finding food [86]. These simulations aimed to model the graceful, but unpredictable, choreography of bird flocks; that is, to model the ability of birds to abruptly, yet synchronously, change their flight direction and then regroup in an optimal formation. From this initial objective, the concept evolved into a simple and efficient stochastic population-based search (optimization) algorithm. In PSO, a collection of individuals, referred to as a swarm of particles, are “flown” through a hyper-dimensional search space. Two very simple principles govern the behaviour of the particles in the swarm. First of all, the particles emulate the success of neighbouring particles, also known as the socio-psychological tendency of the particles. Secondly, particles also emulate their own successes. The collective behavioural pattern that emerges from the swarm is that particles stochastically return toward previously successful regions of the search space; that is, discovering the optimal regions of a high dimensional search space.

Next, core aspects of PSO are discussed in sections 3.2.1.1 to 3.2.1.6. Topics covered include the position and velocity update equations, velocity clamping, the velocity update equation making use of the inertia weight coefficient, the velocity update equation making use of the constriction coefficient, neighbourhood topologies, and parameter sensitivity as related to algorithm stability.

3.2.1.1 Position and Velocity Update Equations

In analogy with the EC paradigm, a swarm is similar to a population and a particle is similar to an individual. Each particle, i, is represented by an n-dimensional vector $\mathbf{x}_i$. The quality of a candidate solution is quantified by evaluating an objective function, $f(\mathbf{x}_i)$. In other words, the objective function measures how "close" each particle is to the optimal objective function value. Each particle moves through the search space by adjusting its position based on the best position encountered by itself, the best position encountered by its neighbours, and its momentum (previous flight direction). This results in the particles converging to the optimum while still exploring the neighbourhoods around the current best position. The position of a particle is updated by adding a velocity, $\mathbf{v}_i$, to its current position. Kennedy and Eberhart [86] defined the particle position and velocity update equations as follows:

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1) \tag{3.4}$$

with

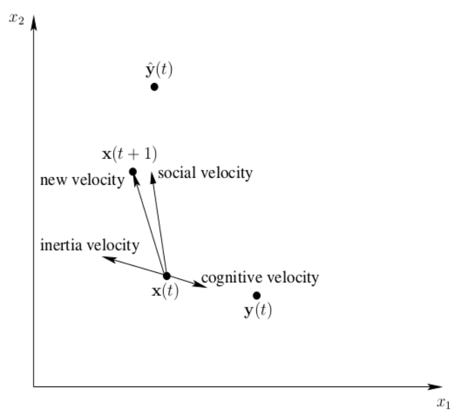
$$\mathbf{v}_i(t+1) = \mathbf{v}_i(t) + c_1\mathbf{r}_1(t) \odot [\mathbf{y}_i(t) - \mathbf{x}_i(t)] + c_2\mathbf{r}_2(t) \odot [\hat{\mathbf{y}}_i(t) - \mathbf{x}_i(t)] \tag{3.5}$$

where $\mathbf{x}_i(t)$ is the position of particle i at iteration t; $\mathbf{v}_i(t)$ is the velocity of particle i at iteration t; c_1 and c_2 are positive acceleration constants, usually in the range $[0,2]$, used to scale the contribution of the second and third term of the velocity equation respectively; $\mathbf{r}_1(t), \mathbf{r}_2(t) \sim U(0,1)^n$ are vectors of random (stochastic) values in the range $[0,1]$, sampled from a uniform distribution; the position and velocity of each particle are respectively initialized as $\mathbf{x}_i(0) \sim U(\mathbf{x}_{min}, \mathbf{x}_{max})^n$ and $\mathbf{v}_i(0) = \mathbf{0}$ [48]; $\mathbf{y}_i(t)$ is the personal best position particle i has visited since the start of the search up until the current iteration t and is initialized to its starting position, i.e., $\mathbf{y}_i(0) = \mathbf{x}_i(0)$; $\hat{\mathbf{y}}_i(t)$ is the best position found by the neighbourhood of particle i since the start of the search up until the current iteration t. The personal and neighbourhood best positions are also referred to as guides since these positions are used in the velocity update equation to guide the particle to better areas of the search space.

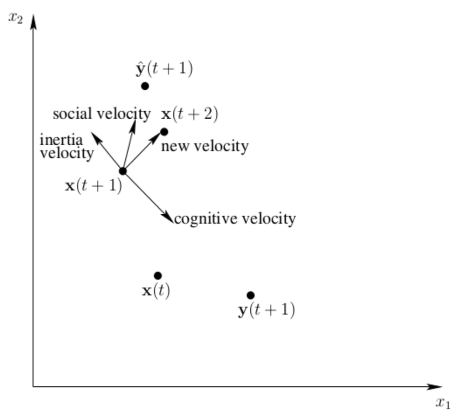
The first term of the velocity Equation (3.5) is also known as the momentum component. The momentum component ensures that a particle is able to maintain some of its previous search direction as the search progresses. The second term of Equation (3.5) is also known as the cognitive component. The cognitive component represents the cumulative experiential knowledge gained by the particle since the start of the search. Finally, the third term of Equation (3.5) is referred to as the social component. The social component represents the information exchanged socially by the particle.

There exist several alternative velocity models, including the cognitive-only, social-only, and selfless models [85]. These velocity model variants elicit different behaviours in the swarm. The cognitive-only model is likened unto nostalgia where particles stochastically return toward their previous best position. With the social-only model, particles only care about the best position of the neighbourhood. For the selfless model a particle also only cares about the best-found neighbourhood position, but without considering its own position. The reader is referred to [85] that discusses the implications when choosing either of the velocity models.

A geometrical illustration of the velocity and position updates for a single two-dimensional particle is shown in figure 3.2 as taken from [47]. Figure 3.2 illustrates an example movement of a particle, where the particle subscript has been dropped for notational convenience. Figure 3.2(a) illustrates the state of the swarm at time step t. Note how the new position, $\mathbf{x}(t+1)$, moves closer towards the global best $\hat{\mathbf{y}}(t)$. For time step $t+1$, as illustrated in figure 3.2(b), assume that the personal best position does not change. The figure shows how the three components contribute to still move the particle towards the global best particle.



(a) Iteration t



(b) Iteration $t+1$

Figure 3.2: Geometrical Illustration of Velocity and Position Updates for a Single Two-Dimensional Particle

3.2.1.2 Velocity Clamping

One of several qualities that characterizes efficient and accurate optimization algorithms is the ability to balance exploration and exploitation during the search. Exploration refers to the ability of a search algorithm to explore different regions of the search space in search of an acceptable optimum [47]. Exploitation, on the other hand, refers to the ability of a search algorithm to concentrate the search around a promising region with the goal of improving a candidate solution [47]. Within the PSO algorithm, these objectives are addressed by the velocity update equation, which drives the search process.

The velocity update in Equation (3.5) has three terms (momentum, cognitive, and social) that each contribute to the step size of particles. When a particle, $\mathbf{x}_i(t)$, is distant from its personal best position, $\mathbf{y}_i(t)$, and/or its neighbourhood best position, $\hat{\mathbf{y}}_i(t)$, it may cause the velocity of that particle to "explode", causing the velocity to increase towards infinity [157]. Part of the problem is also that the degree to which the inertia component, in Equation (3.5), influences the search is left uncontrolled.

A particle with an extreme velocity is undesirable because the particle will then experience a large position update causing it to roam outside the search space, failing to return to the search space [50, 70]. This problem is intensified when it happens for multiple particles, causing the swarm to diverge [50, 70].

Eberhart and Kennedy proposed clamping the velocity of the particles [46] to control the roaming behaviour of particles. The effect of velocity clamping is that particles whose velocities exceed some upper or lower limit are adjusted to adhere to the "speed limit". Note that velocity clamping is applied to each velocity vector post the velocity update equation, but prior to being used in the position update equation.

There are different variations of velocity clamping but the simplest approach is to set a static minimum and maximum velocity value in each dimension. When one lacks prior knowledge about the problem being solved, Shi and Eberhart [45, 145] suggested controlling the global exploration of the PSO algorithm by setting the allowed velocity range proportional to the boundary constraints. The optimal proportion, however, is problem-dependent [145]. Let $\mathbf{v}_{max}$ be the maximum velocity. Then the velocity update equation, incorporating velocity clamping, is of the form:

$$\mathbf{v}_i(t+1) = \begin{cases} \mathbf{v}_i(t+1) & \text{if } -\mathbf{v}_{max} \leq \mathbf{v}_i(t+1) \leq \mathbf{v}_{max} \\ \mathbf{v}_{max} & \text{if } \mathbf{v}_i(t+1) > \mathbf{v}_{max} \\ -\mathbf{v}_{max} & \text{if } \mathbf{v}_i(t+1) < -\mathbf{v}_{max} \end{cases} \tag{3.6}$$

Note that velocity clamping does not restrict the positions of particles, only the step sizes as determined by the velocity of each particle. However, velocity clamping also influences the direction in which a particle moves, which can be said to improve exploration but can also cause the optimum to not be found. Another possible pitfall of velocity clamping is when all the velocities are equal to the maximum velocity, which causes the particles to only search on the boundary of the hypercube defined by the velocity constraints. This will hinder the effectiveness of the PSO algorithm since the global optimum is probably not located on the boundary of the search space. Note that velocity clamping is optional when the control parameters w, c_1, and c_3 are set to values that ensure that the swarm will converge to some set of solutions, i.e., the particles will stop moving. These values can be determined using theoretically derived stability conditions [17]. Note that other dynamic velocity mechanisms have also been used [76, 140].

3.2.1.3 Particle Swarm Optimization with Inertia Weight Coefficient

The momentum component of the velocity update equation causes each particle to maintain its current trajectory. This may cause particles to oscillate between positions and become stuck. This can also cause particles to move through areas of the search space that may not be directly on coarse to previously observed successes, thereby encouraging exploration of the search space [144]. There must be a balance between exploration and exploitation: a swarm that only explores and never refines good solutions will waste resources exploring fruitless areas of the search space. A swarm that only exploits is likely to converge prematurely to a local optimum. Shi and Eberhart [144] introduced the inertia weight, w, as a mechanism to control the exploration and exploitation of a swarm. In other words, the inertia weight controls the momentum of each particle; that is, how much the previous flight direction of a particle influences its new velocity. The velocity update equation of the PSO algorithm, incorporating the inertia weight coefficient, is defined as

$$\mathbf{v}_i(t+1) = w\mathbf{v}_i(t) + c_1\mathbf{r}_1(t) \odot [\mathbf{y}_i(t) - \mathbf{x}_i(t)] + c_2\mathbf{r}_2(t) \odot [\hat{\mathbf{y}}_i(t) - \mathbf{x}_i(t)] \quad (3.7)$$

where $w \in (0,1)$.

Note that the inertia weight does not replace the need for velocity clamping when unstable control parameter values are used [154, 157, 158]. A large inertia weight will cause extreme particle velocities due to the cumulative effect over time [157]. For $w \geq 1$, velocities increase over time, towards the maximum velocity or infinity depending on whether velocity clamping is used or not [157]. For $w < 1$, particles decelerate until they are motionless (only if the stability conditions are satisfied) [157]. In other words, a small inertia weight promotes local exploitation, whereas, a large inertia weight promotes exploration and diversity (depending on the values of c_1 and c_2).

The inertia weight, like all the PSO algorithm parameters in general, is problem-dependent. A static inertia weight can be used for the entire duration of the search, or one of several dynamic methods [22, 88, 110, 113, 119, 120, 127, 147, 149, 163, 164, 176] can be used to update the inertia weight as the search progresses.

The pseudocode for a generic PSO algorithm, incorporating the inertia weight, is presented in Algorithm 5. Note that Algorithm 5 is for a synchronous PSO algorithm where all the personal best and neighbourhood best positions are updated prior to updating the particle velocities and positions.

When choosing an appropriate stopping condition it is important to protect against premature convergence to sub-optimal solutions as well as unnecessary computational overhead. One way to protect against premature convergence when choosing one or more stopping conditions is to allow the algorithm enough execution time (iterations or objective function evaluations) to be able to explore the search space to find a good solution [47]. Other alternatives include stopping when a solution with an acceptable error has been found, no improvement is observed over a number of iterations, the normalized swarm radius is close to zero, and/or the objective function slope is approximately zero [47].

3.2.1.4 Particle Swarm Optimization with Constriction Coefficient

Bypassing the need for velocity clamping, Clerc and Kennedy [21, 23] developed a constriction factor (coefficient), χ, to control the step size of the velocity of a particle. The constriction factor ensures a convergent swarm state, i.e., $\mathbf{v}_i(t) = \mathbf{0}, \forall i = 1, ..., n_s$. The resulting PSO velocity update equation with the constriction factor incorporated is defined as

$$\mathbf{v}_i(t+1) = \chi\big(w\mathbf{v}_i(t) + c_1\mathbf{r}_1(t) \odot [\mathbf{y}_i(t) - \mathbf{x}_i(t)] + c_2\mathbf{r}_2(t) \odot [\hat{\mathbf{y}}_i(t) - \mathbf{x}_i(t)]\big) \quad (3.8)$$

with

$$\chi = \frac{2}{2-\varrho-\sqrt{\varrho^2-4\varrho}} \quad (3.9)$$

where

$$\varrho = \begin{cases} c_1 + c_2 & \text{if } c_1 + c_2 > 4 \\ 0 & \text{if } c_1 + c_2 \leq 4 \end{cases}$$

Small values for the constriction factor, i.e. $\chi \approx 0$, encourage exploitation and faster convergence. Larger values for the constriction factor, i.e. $\chi \approx 1$, encourage exploration and slow convergence.

3.2.1.5 Neighbourhood Topologies

The power of PSO resides in the ability of simple particles to share information in order to facilitate a complex and intelligent behaviour exhibited by the swarm as a whole. The neighbourhood topology (social structure) of a particle describes this information exchange among the other particles in the swarm.

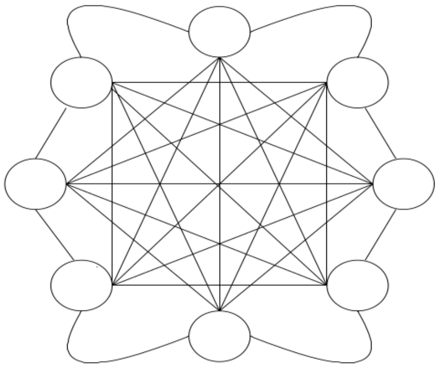
Algorithm 5 Particle Swarm Optimization (PSO)

```
Create and initialize an n-dimensional swarm, S, to consist of
n_s particles;
Let t = 0;
while stopping condition(s) not true do
   for each particle i = 1, ..., n_s do
      if f(x_i(t)) < f(y_i) then
         y_i = x_i(t); // set personal best position
      end if
      for particles î with particle i in their neighbourhood do
         if f(y_i) < f(ŷ_î) then
            ŷ_î = y_i; // set neighbourhood best position
         end if
      end for
   end for
   for each particle i = 1, ..., n_s do
      Update the velocity using Equation (3.7);
      Update the position using Equation (3.4);
   end for
   t = t + 1;
end while
```

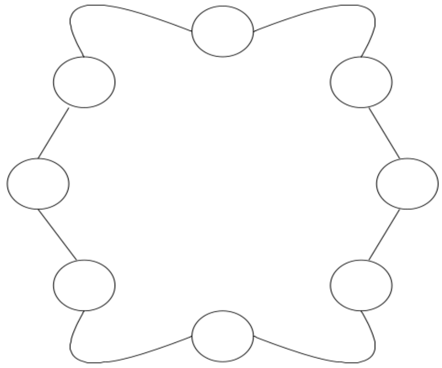
A topology can be visualized by a graph where the vertices represent the particles and the edges represent the "communication channels". A number of neighbourhood topologies exist, with the two most popular and extreme being the star and ring topologies. The star and ring social structures are shown in figure 3.3 as taken from [47]. Other social structures are discussed in [47].

The star or fully connected neighbourhood in figure 3.3 (a) connects each particle in the swarm with every other particle in the swarm. The entire swarm, therefore, belongs to a single neighbourhood. This allows for rapid exploitation of the available information regarding promising particle positions, which leads to faster convergence and less exploration [49]. PSO algorithms that use the star social structure are referred to as global best (gbest) PSO algorithms.

The ring topology restricts the flow of information to some number of the immediate neighbours of a particle; that is, the neighbourhood size. In fig-



(a) Star Topology



(b) Ring Topology

Figure 3.3: The Star and Ring Neighbourhood Topologies

ure 3.3 (b), each particle can communicate with its two adjacent particles (i.e. a neighbourhood size of two). Note that the particles communicate in the abstract sense, i.e., through the velocity update in Equation (3.7). The swarm has several overlapping neighbourhoods, allowing for gradual information exchange between all the particles, which results in a slower convergence rate and greater exploration. The star topology can be seen as a special case of the ring topology where the neighbourhood size is equal to the size of the swarm minus one. PSO algorithms that use the ring topology are known as local best (lbest) PSO algorithms.

As with other PSO parameters, the optimal neighbourhood topology is problem-dependent. Engelbrecht [49] showed that neither gbest nor lbest PSO is superior to the other in any problem class.

Note that, in general, the neighbourhood of a particle is determined by its index and not its position in the search space. This index (non-spatial) approach promotes the flow of information between different neighbourhoods of the actual search space since neighbourhoods can overlap. Spatial topologies, on the other hand, are computationally more expensive than their non-spatial counterparts since these involve the calculation of the Euclidean distance between all pairs of particles. Spatial neighbourhoods can also cause particles to fruitlessly roam bad neighbourhoods. This unwanted trait is due to the fact that these particles become trapped in undesirable regions because they do not receive information about superior solutions outside of their immediate spatial neighbourhood.

3.2.1.6 Parameter Sensitivity and Algorithm Stability

The PSO algorithm is sensitive to the assigned parameter values [6, 85, 121, 145, 146, 157], which include the swarm size, neighbourhood size, inertia weight, and acceleration coefficients. The value of w, c_1, and c_2 is extremely important to ensure convergent (i.e. stable) swarm behaviour. When choosing a value for w, the values selected for c_1 and c_2 should also be taken into account. Typically, the values for c_1 and c_2 are chosen in the range $[0, 2]$ and that of w is in $(0, 1)$. Although some regard PSO to be fairly robust concerning the selection of these parameters [46], infelicitous values may lead to premature convergence, excessive roaming, or cause the swarm to exhibit divergent or cyclic behaviour [121, 157, 158].

Eberhart and Shi [46] proposed "rule of thumb" parameter values for which the PSO algorithm performs well. The chosen values were $c_1 = c_2 = 1.49445$ and $w = 0.7929$. However, later theoretical research has shown that there exists a convergent region within which swarm convergence to an equilibrium state is guaranteed [18, 19, 23, 62, 82, 84, 114, 115, 121, 122, 154, 157, 158, 170, 176]. Equilibrium is reached when the entire swarm stops moving (i.e. the velocity of all particles become zero). Algorithm stability, therefore, refers to the ability of the swarm to converge to an equilibrium state. For stable parameter values, those satisfying the stability conditions, it is guaranteed that the expectation and variance of particle positions will converge to constant values (i.e. the particles become motionless).

Cleghorn and Engelbrecht [19] generalized the work done by Van den Bergh and Engelbrecht [158], Van den Bergh [157], and Trelea [154] and found that convergent particle trajectories are guaranteed for parameter values that satisfy the following condition:

$$c_1 + c_2 < 2(1 + w), \quad \text{for } -1 < w < 1,\ c_1 > 0 \text{ and } c_2 > 0 \tag{3.10}$$

Gazi [62] expanded the region derived by Kadirkamanathan *et al.* [84] and found that convergence is guaranteed for parameter values that satisfy the following condition:

$$\begin{aligned} c_1 + c_2 &< \tfrac{24(1+w)}{7}, \quad \text{for } -1 < w \le 0 \\ c_1 + c_2 &< \tfrac{24(1-w)^2}{7(1+w)}, \quad \text{for } 0 < w \le 1 \end{aligned} \tag{3.11}$$

The most accurate convergent region (stability condition) was independently derived by Poli [121], Poli and Broomhead [122], and Jiang *et al.* [82], given below:

$$c_1 + c_2 < \tfrac{24(1-w)^2}{7-5w}, \quad \text{for } -1 \le w \le 1 \tag{3.12}$$

Other, less accurate, approaches exist that vary the inertia weight and acceleration coefficients over time or based on other information of the search space [66, 117, 162]. Note that these are not stability conditions.

3.2.2 Many-objective Particle Swarm Optimization

PSO algorithms, as with EAs, has also matured since the first of its kind and has been successfully applied to MaOPs. Section 3.2.2.1 gives an overview of yet another benchmark MaOO algorithm used in this study. Next, a detailed overview of the focal algorithm of this study, the MGPSO algorithm, is given in section 3.2.2.2.

3.2.2.1 Controlling Dominance Area of Solutions Speed Constraint Multi-objective Particle Swarm Optimization

The speed constraint multi-objective particle swarm optimization (SMPSO) algorithm [112] uses the constriction coefficient (discussed in section 3.2.1.4) developed by Clerc and Kennedy [21, 23] to limit the velocity of the swarm. For each iteration, the parameters of the SMPSO algorithm are varied randomly in some range which, together with the constriction coefficient, guarantees a stable state, i.e., $\mathbf{v}_i(t) = \mathbf{0}, \forall i = 1, ..., n_s$ [21, 23, 112]. The SMPSO algorithm, in addition to the constriction coefficient, also uses a mechanism to further bound the accumulated velocity of each dimension, in each particle. Carvalho and Pozo [33] combined the SMPSO algorithm with the controlling dominance area of solutions (CDAS) [32, 132] ranking method, referred to as the CDAS-SMPSO algorithm.

An archive, $\mathcal{A}$, is a "container" used to store promising non-dominated solutions found throughout the search. The CDAS-SMPSO algorithm uses a bounded archive; that is, an archive with a fixed capacity. The alternative to a bounded archive is an unbounded archive, which has no limit on the number of solutions it can store. An unbounded archive is seldomly practical due to storage limitations and the computational costs associated with maintaining large archives.

The CDAS method, as its name suggests, controls the levels of contraction and expansion of the dominance area of particles in the swarm. Contraction and expansion are proportional to a user-defined control parameter, γ, that can be defined for each objective m. However, this study uses the same γ-value for each objective as done in [106]. Note that the optimal value for γ is problem-dependent. The goal with the CDAS technique is to induce an appropriate non-dominated ranking of the particles by modifying the objective function values of each objective. Appropriate here refers to the ranking that will best guide the swarm to the POF. The objective function value modification is done by using a trigonometric operation that modifies the objective function value of each particle into a new objective value. The objective function value modification is necessary to be able to induce a stronger or weaker selection pressure depending on the problem and the value of γ to help guide the swarm towards the true POF. The modification of the dominance area is defined as

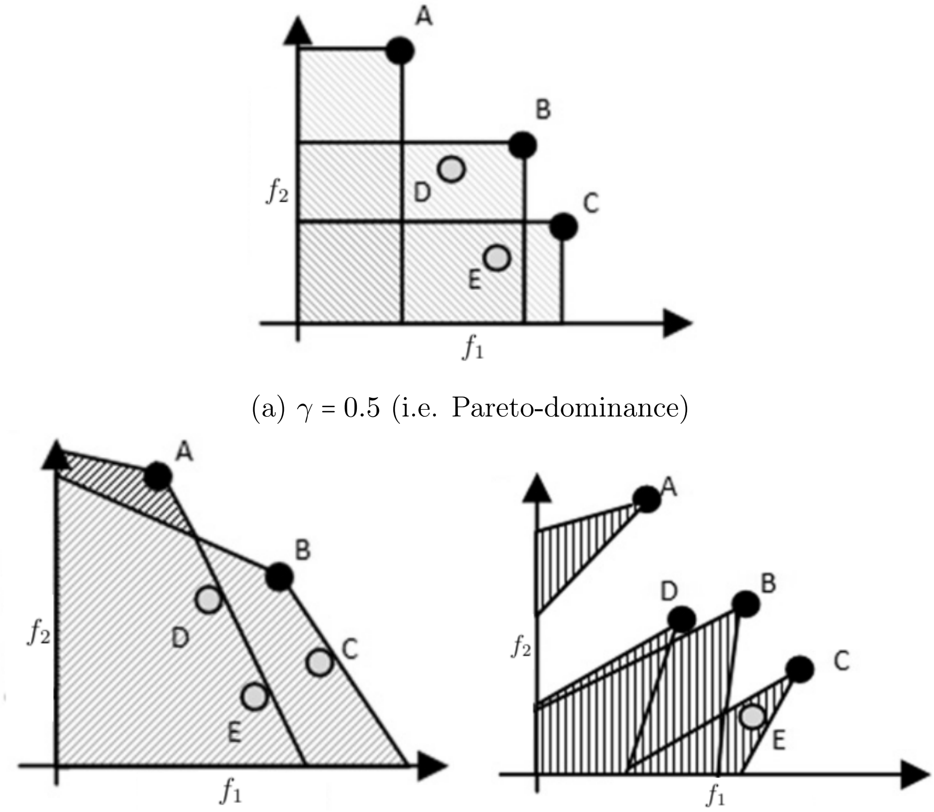
$$f''_m(\mathbf{x}) = \frac{||\mathbf{f}(\mathbf{x})||_2 \cdot \sin(\omega_m + \gamma\pi)}{\sin(\gamma\pi)} \tag{3.13}$$

where

$$\omega_m = \arccos\left(\frac{f_m(\mathbf{x})}{||\mathbf{f}(\mathbf{x})||_2}\right)$$

and $f''_m(\mathbf{x})$ is the modified function value for objective m; $\|\mathbf{f}(\mathbf{x})\|_2$ is the Euclidean norm of $\mathbf{f}(\mathbf{x})$; and ω_m is the declination angle between $f_m(\mathbf{x})$ and $\mathbf{f}(\mathbf{x})$.

Figure 3.4 depicts the modified dominance area of solutions associated with different values for γ. Note that figure 3.4 represents a MOP requiring the maximization of two objectives. For $\gamma = 0.5$, shown in figure 3.4 (a), the new fitness value is equal to the old fitness value; that is, an unmodified dominance area. For $\gamma < 0.5$, shown in figure 3.4 (b), fewer particles will become non-dominated and the particles tend to converge to positions that translate to a small area of the objective space (i.e. promoting exploitation). For $\gamma > 0.5$, shown in figure 3.4 (c), some particles that were dominated will become non-dominated and the algorithm tends to find a more diverse set of solutions (i.e. promoting exploration).



(a) $\gamma = 0.5$ (i.e. Pareto-dominance)

(b) $\gamma < 0.5$

(c) $\gamma > 0.5$

Figure 3.4: Illustration of CDAS for a maximization MOP with two objectives.

The CDAS technique was incorporated in the CDAS-SMPSO algorithm search as follows: the CDAS technique modifies the objective function values according to Equation (3.13) whenever the Pareto-dominance relation is applied (i.e. whenever the archive is updated). This modification directly influences the neighbourhood guide selection step of the CDAS-SMPSO algorithm. With the CDAS method, a subset of non-dominated solutions can be selected or a relaxed update of the archive can be performed.

Algorithm 6 gives the pseudocode for the CDAS-SMPSO algorithm. It starts by initializing a swarm (line 1), which includes the position, velocity, and personal best of the particles. The archive is initialized with the non-dominated solutions in the swarm (line 2). The CDAS-SMSPSO algorithm stores non-dominated solutions in a bounded archive. When the archive reaches maximum capacity (usually the swarm size), the crowding distance operator [35] is used to determine which solutions are allowed to remain. That is, the least crowded solutions are safe.

Then, the main loop of the algorithm is executed for a maximum number of iterations. The constricted velocities [21, 23] of the particles are calculated first (line 5). The neighbourhood best position of a particle, part of the velocity update equation, is selected as the winner of a random binary tournament. Two particles are randomly selected from the archive and the particle with the largest crowding distance to its nearest neighbours in the archive is selected to be the winner. Next, the positions of the particles are calculated (line 6) and a mutation operator is applied with a given probability (line 7). Mutation is a concept borrowed from the EC paradigm that is used to introduce more diversity. The addition of a mutation operator can also be said to promote exploration of other regions of the search space. The archive is then updated with the newly discovered non-dominated particle positions, as determined by applying the Pareto-dominance operator to the CDAS-modified objective function values (line 8). Finally, just before advancing to the next iteration, the personal best position of each particle is updated if its current position dominates its personal best position or if neither position dominates the other; that is, both positions are non-dominated with respect to each other (line 9). Note that when updating personal best positions the unmodified objective function values are considered. As the search concludes, the algorithm returns the archive as the found (approximated) POF (line 12).

The CDAS-SMPSO algorithm has been shown to perform well on MaOPs [33, 106]. Therefore, it was also included as part of the benchmark algorithms used in this study. The reader is referred to [33] for greater detail about the CDAS-SMPSO algorithm.

3.2.2.2 Multi-guide Particle Swarm Optimization

The MGPSO algorithm was developed by Scheepers *et al.* [137]. The MGSPO algorithm is a multi-swarm multi-objective PSO algorithm that utilizes multiple swarms (subswarms), one per objective. Note that the first multi-swarm multi-objective PSO algorithm was the vector evaluated particle swarm optimization (VEPSO) algorithm [117, 135]. Each subswarm is dedicated to optimizing only one objective, independent of the other objectives. Particles

Algorithm 6 Controlling Dominance Area of Solutions Speed Constraint Multi-objective Particle Swarm Optimization (CDAS-SMPSO)

1 : Create and initialize an n-dimensional swarm, S, to consist of n_s particles;
2 : Initialize the archive, $\mathcal{A}$;
3 : Let $t = 0$;
4 : **while** stopping condition(s) not true **do**
5 :   Compute the constricted particle velocities;
6 :   Update particle positions;
7 :   Apply mutation (turbulence);
8 :   Update the archive;
9 :   Update personal best positions;
10 : $t = t + 1$;
11 : **end while**
12 : **return** $\mathcal{A}$;

in each subswarm are evaluated using the objective function associated with that subswarm. The personal and neighbourhood best positions of the MGPSO algorithm are also updated according to the objective function value of the corresponding objective function.

The MGPSO algorithm adds an archive guide term to the velocity update equation to facilitate information exchange between the subswarms. The exchanged information helps the subswarms to find promising solutions with respect to all of the objectives of the problem. The MGPSO velocity update equation is defined as

$$\begin{aligned} \mathbf{v}_i(t+1) &= w\mathbf{v}_i(t) + c_1\mathbf{r}_1(t) \odot [\mathbf{y}_i(t) - \mathbf{x}_i(t)] + \lambda_i c_2 \mathbf{r}_2(t) \odot [\hat{\mathbf{y}}_i(t) - \mathbf{x}_i(t)] \\ &\quad + (1-\lambda_i)c_3\mathbf{r}_3(t) \odot [\hat{\mathbf{a}}_i(t) - \mathbf{x}_i(t)] \end{aligned} \tag{3.14}$$

where c_3 is an acceleration coefficient; $\mathbf{r}_3(t)$ is a random vector with components sampled uniformly from $(0, 1)$; $\hat{\mathbf{a}}_i(t)$ is the randomly selected archive guide for particle i at iteration t; λ_i is the archive balance coefficient for particle i.

The archive balance coefficient controls the amount of influence that the archive guide has on the velocity of the particle. In other words, the archive balance coefficient controls how much the already found POF (stored in an archive) is exploited. Large values for the archive balance coefficient decreases the influence of the archive guide and simultaneously increase the influence of the neighbourhood guide. Small values for the archive balance coefficient does the opposite, increasing the influence of the archive guide while decreasing the influence of the neighbourhood guide. The MGPSO uses a bounded archive.

The archive guide is selected using tournament selection. Three solutions are randomly selected to partake in the tournament and the winner is chosen to be the particle with the largest corresponding crowding distance. Selection of the winner of the tournament using crowding distance encourages the MGPSO algorithm to prioritize the exploration of sparsely populated areas over densely populated areas of the objective space, i.e., to promote global exploration and diversity [136].

At each iteration, each particle is allowed to deposit its current position into the archive if it is non-dominated with respect to the solutions already inserted into the archive. When a non-dominated solution has to be inserted into the archive, and there is still space available, the solution is simply inserted. Otherwise, if the archive is full, the new non-dominated particle position replaces the most crowded solution in the archive - in doing so, the diversity of the found POF is preserved [136]. After a new solution has been inserted into the archive, any solutions in the archive that is then dominated by the newly inserted solution are removed. The pseudocode for archive insertion in the MGPSO algorithm is presented in Algorithm 7.

Algorithm 7 Multi-guide Particle Swarm Optimization (MGPSO) - Archive Insert

```
if solution to be inserted is non-dominated do
    if space available in 𝒜 do
        Insert non-dominated solution into 𝒜;
        Remove any dominated solutions from 𝒜;
    end if
    else if 𝒜 has reached maximum capacity do;
        Remove most crowded solution from 𝒜;
        Insert non-dominated solution into 𝒜;
        Remove any dominated solutions from 𝒜;
    end if
end if
return 𝒜;
```

Archive balance coefficients are sampled per particle randomly from a uniform distribution in the range $[0, 1]$. The values assigned to the archive balance coefficients do not change. This original approach is referred to as the standard static archive balance coefficient update strategy (STD).

Erwin and Engelbrecht [53] investigated the use of five different dynamic archive balance coefficient update strategies. The dynamic approaches change the value of the archive balance coefficient at each iteration of the search. Each of the five dynamic archive balance coefficient update strategies is defined and discussed below.

The linearly decreasing dynamic archive balance coefficient update strategy (LD) initializes the archive balance coefficient of each particle to 1.0. Thereafter, λ_i is updated as

$$\lambda_i(t+1) = \lambda_i(t) - \frac{1.0}{n_t} \tag{3.15}$$

where n_t is the maximum number of iterations.

The linearly increasing dynamic archive balance coefficient update strategy (LI) initializes the archive balance coefficient of each particle to 0.0. Thereafter, λ_i is updated as

$$\lambda_i(t+1) = \lambda_i(t) + \frac{1.0}{n_t} \tag{3.16}$$

The LI approach gradually increases the influence of the neighbourhood guide, while decreasing the influence of the archive guide, whereas, the LD approach does the opposite.

Three random dynamic archive coefficient update strategies were investigated, each with a different degree of stochasticity:

- Coarse granularity: The random dynamic archive balance coefficient update strategy (R) samples a new archive balance coefficient at every iteration. This value is used by all particles. Thus, every particle will have the same archive balance coefficient at each iteration, updated as

$$\lambda_i(t) = \lambda(t) \sim U(0,1), \forall i = 1, ..., n_s \tag{3.17}$$

- Medium granularity: The random update per particle dynamic archive balance coefficient update strategy (RI) randomly assigns a new archive balance coefficient to each particle, at every iteration. The archive balance coefficient is updated as

$$\lambda_i(t) \sim U(0,1) \tag{3.18}$$

- Fine granularity: The random update per particle per dimension (RIJ) strategy randomly assigns a new archive balance coefficient to every dimension $j_n \in \{1, ..., n\}$ of each particle, at every iteration. In this case, the archive balance coefficient is updated as

$$\lambda_{ij_n}(t) \sim U(0,1) \tag{3.19}$$

The linearly increasing strategy and stochastic strategies have been shown to outperform the standard approach to initializing the archive balance coefficient on optimization problems with two and three objectives [53].

As expected, the performance of the MGPSO algorithm is also sensitive to the values assigned to its parameters [136]: poorly chosen parameter values can cause the MGPSO algorithm to diverge (i.e. finding no solution at all). It is extremely tedious and resource-intensive to find the optimal combination of parameter values for a specific problem. Stability conditions of the MGPSO algorithm have been derived [20, 137]. This means that initializing the MGPSO algorithm with parameter values for w, c_1, c_2, c_3, and λ that satisfy the stability conditions guarantees that the algorithm will converge, i.e., all the particles will eventually stop moving. The stability conditions for the MGPSO algorithm, with differing acceleration coefficient values [137], are defined as

$$0 < c_1 + \lambda c_2 + (1 - \lambda)c_3 < \frac{4(1-w^2)}{1-w+\frac{(c_1^2+\lambda^2 c_2^2+(1-\lambda)^2 c_3^2)(1+w)}{3(c_1+\lambda c_2+(1-\lambda)c_3)^2}}, \ |w| < 1 \tag{3.20}$$

Detailed pseudocode for the MGPSO algorithm is presented in Algorithm 8.

Algorithm 8 Multi-guide Particle Swarm Optimization (MGPSO) - Search

Perform algorithm initialization (Algorithm 9);
Let $t = 0$;
while stopping condition(s) not true **do**
  for each objective $m = 1, \ldots, n_m$ **do**
    for each particle $i = 1, \ldots, S_m.n_{s_m}$ **do**
      if $f_m(S_m.\mathbf{x}_i(t)) < f_m(S_m.\mathbf{y}_i)$ **then**
        $S_m.\mathbf{y}_i = S_m.\mathbf{x}_i(t)$; // set personal best position
      end if
      for particles $\hat{\imath}$ with particle i in their neighbourhood **do**
        if $f_m(S_m.\mathbf{y}_i) < f_m(S_m.\hat{\mathbf{y}}_{\hat{\imath}})$ **then**
          $S_m.\hat{\mathbf{y}}_{\hat{\imath}} = S_m.\mathbf{y}_i$; // set neighbourhood best position
        end if
      end for
      Update the archive, $\mathcal{A}$, with the solution $S_m.\mathbf{x}_i(t)$ (Algorithm 7);
    end for
  end for
  for each objective $m = 1, \ldots, n_m$ **do**
    for each particle $i = 1, \ldots, S_m.n_{s_m}$ **do**
      Select a solution, $S_m.\hat{\mathbf{a}}_i(t)$, from the archive using tournament selection;

$$\begin{aligned} S_m.\mathbf{v}_i(t+1) = w S_m.\mathbf{v}_i(t) & \\ &+ c_1(t)\mathbf{r}_1(t) \odot [S_m.\mathbf{y}_i(t) - S_m.\mathbf{x}_i(t)] \\ &+ S_m.\lambda_i(t) c_2(t)\mathbf{r}_2(t) \odot [S_m.\hat{\mathbf{y}}_i(t) - S_m.\mathbf{x}_i(t)] \\ &+ (1 - S_m.\lambda_i(t)) c_3(t)\mathbf{r}_3(t) \odot [S_m.\hat{\mathbf{a}}_i(t) - S_m.\mathbf{x}_i(t)]; \end{aligned}$$

      $S_m.\mathbf{x}_i(t+1) = S_m.\mathbf{x}_i(t) + S_m.\mathbf{v}_i(t+1)$;
    end for
  end for
  $t = t + 1$;
end while
return $\mathcal{A}$;

Algorithm 9 Multi-guide Particle Swarm Optimization (MGPSO) - Initialization

for each objective $m = 1, ..., n_m$ **do**
    Create and initialize a swarm, S_m, of n_{s_m} particles uniformly within a predefined hypercube of dimension n;
    Let f_m be the objective function;
    Let $S_m.\mathbf{y}_i$ represent the personal best position of particle $S_m.\mathbf{x}_i$, initialized to $S_m.\mathbf{x}_i(0)$;
    Let $S_m.\hat{\mathbf{y}}_i$ represent the neighbourhood best position of particle $S_m.\mathbf{x}_i$, initialized to $S_m.\mathbf{x}_i(0)$;
    Initialize $S_m.\mathbf{v}_i(0)$ to 0;
    Initialize $S_m.\lambda_i \sim U(0, 1)$;
end for

3.3 Summary

This chapter gave an overview of, and a justification for, the many-objective optimization (MaOO) algorithms used in this study. More specifically, this chapter dealt with the two types of algorithms, i.e., evolutionary algorithms (EAs) and particle swarm optimization (PSO) algorithms. The basic single-objective EA and PSO algorithm were introduced first before discussing the multi- and many-objective variants that were used in this study.

The many-objective evolutionary algorithms (MaOEAs) used in this study, each with their most prominent features, are listed below:

- The knee-point driven evolutionary algorithm (KnEA) [174], which incorporates knee-points as a secondary convergence metric to aid the Pareto-dominance relation.
- The many-objective evolutionary algorithm based on dominance and decomposition (MOEA/DD) [95], which decomposes the MaOP into scalar optimization problems to be optimized cooperatively and uses the non-dominance concept. The MOEA/DD also uses a set of weight vectors (reference-points) to help guide the search.
- The reference-point based many-objective non-dominated sorting genetic algorithm (NSGA-III) [38], which favours non-dominated solutions that are also near a specified reference-front.

The many-objective particle swarm otpimization (MaOPSO) and multi-objective particle swarm otpimization (MOPSO) algorithms used in this study, each with their most prominent features, are listed below:

- The controlling dominance area of solutions speed constraint multi-objective particle swarm optimization (CDAS-SMPSO) algorithm [33], which modifies the Pareto-dominance relation with the CDAS method to induce an appropriate ranking of the swarm.
- The multi-guide particle swarm optimization (MGPSO) algorithm [137], an aspiring MaOPSO algorithm, which uses the Pareto-dominance relation.

As mentioned, the MGPSO uses the original Pareto-dominance relation and is therefore expected to not scale well from multi-objective optimization problems (MOPs) to many-objective optimization problems (MaOPs) [68, 77, 81, 106, 141]. The two novel, hopefully scalable, MGPSO algorithm adaptations are presented in the chapters to follow, together with the empirical results.

Chapter 4

Partial-dominance Multi-guide Particle Swarm Optimization

"Most people are more comfortable with old problems than with new solutions."

— Charles H. Bower

This chapter proposes a new variation of the MGPSO algorithm, i.e., the partial-dominance multi-guide particle swarm optimization (PMGPSO) algorithm. As sub-objectives, this chapter aims to: evaluate the performance of the MGPSO algorithm and PMGPSO algorithm in comparison with other algorithms; evaluate the effect of different archive balance coefficient update strategies for the PMGPSO algorithm; investigate if the scalability of the MGPSO algorithm can be improved by using different dynamic archive balance coefficient update strategies. This chapter also empirically compares the scalability of the various MOO and MaOO algorithms. More specifically, section 4.1 proposes partial-dominance as an approach to improve the scalability of the MGPSO algorithm. The PMGPSO algorithm is presented in section 4.2, the empirical process followed in this study is presented in section 4.3, and section 4.4 presents the results and discusses the findings. Finally, section 4.5 gives a summary of this chapter.

4.1 Partial-dominance Approach

Helbig and Engelbrecht [52, 69] proposed the partial-dominance approach to replace the Pareto-dominance relation. Whenever two solutions need to be compared, the partial-dominance relation randomly selects three of the objectives and applies the Pareto-dominance relation only on these three distinct objectives. The partial-dominance approach can be thought of as an adaptation of the Pareto-dominance relation. The partial-dominance approach can also be considered a temporary random objective reduction approach, because whenever two solutions are compared, to determine which is best, this approach randomly selects three objectives while neglecting the other objectives of the MaOP. The partial-dominance relation selects objectives randomly, with an equal selection probability for each objective. For the partial-dominance relation, the probability of objective m being selected is calculated as

$$P_m^{PD} = \frac{1}{n_m} \tag{4.1}$$

The partial-dominance relation was applied to the NSGA-II [35] and a MOPSO [129] algorithm, and shown to scale these multi-objective algorithms to many-objectives [52, 69]. Therefore, this study proposes to investigate if the partial-dominance relation will help the MGPSO algorithm to scale.

4.2 Partial-dominance Multi-guide Particle Swarm Optimization

Recall from section 3.2.2.2 that the MGPSO algorithm only uses the Pareto-dominance relation during the archive update phase. The Pareto-dominance relation is known for struggling to effectively identify desirable solutions as the number of objectives increases [77, 106, 141]. As the number of objectives increases the size of the objective space also increases. This, in turn, results in most solutions becoming non-dominated early on and will guide the particles to sub-optimal regions of the search space. Recall from section 2.2.2 that the Pareto-dominance relation lacks selection pressure towards the POF when used as the sole convergence-related metric. Therefore, the performance of the MGPSO algorithm, and other pure Pareto-based MOO algorithms for that matter, heavily depends on whether solutions that can promote convergence towards, and diversity along, the POF will be selected to guide the search during MaOO. Unfortunately, since no other convergence-related metric is used, the diversity-related operator largely guides the search, resulting in most algorithms terminating with a set of diverse sub-optimal solutions [1, 81, 165]. That is, the search is lead by dominance-resistant solutions that are extremely inferior to others in at least one objective but hardly ever dominated [77, 105].

The PMGPSO algorithm is the same as the original MGPSO algorithm except for the differences discussed next. The PMGPSO algorithm is an adaptation of the original MGPSO algorithm. The PMGPSO algorithm proposes to use the partial-dominance relation instead of the Pareto-dominance relation with the goal of scaling the MGPSO algorithm to many-objectives. The partial-dominance relation, used by the PMGPSO algorithm, belongs to the category of methods which modifies the Pareto-dominance relation in order to increase the selection pressure. Partial-dominance can also be said to belong to the category of methods which reduces the dimensionality of the problem because of how the relation randomly selects a subset of objectives each time the relation is applied. The PMGPSO algorithm also aims to find a set of solutions well-spread (diverse) along the true POF by using the diversity promoter and preserver (the crowding distance operator) already utilized by the original MGPSO.

The PMGPSO algorithm replaces the Pareto-dominance relation used by the original MGPSO algorithm with the partial-dominance relation. In other words, in Algorithm 7 the PMGPSO algorithm uses the partial-dominance relation instead of the Pareto-dominance relation when attempting to insert a solution into the archive. By doing this, the archive no longer stores fully non-dominated solutions (when considering all of the objectives), instead partially-dominated solutions are stored which might be dominated by other solutions in some of the objectives but are non-dominated for at least the three randomly selected objectives. As a consequence of storing partially-dominated solutions, the PMGPSO algorithm does not remove dominated solutions from the archive after inserting a solution into the archive (lines 4 and 9 of Algorithm 7). That is, the MGPSO algorithm removes any newly dominated solutions from the archive after inserting a non-dominated solution, whereas the PMGPSO algorithm simply inserts a partially-dominated solution and then continues with the next step of the algorithm. Note that for three objectives, the Pareto-dominance relation is identical to the partial-dominance relation. However, for three objectives the MGPSO and PMGPSO algorithms still differ with regards to lines 4 and 9 of Algorithm 7 as explained above. In the case of a full archive, the new partially-dominated particle position replaces the most crowded solution in the archive, thereby preserving the diversity of the found POF [136].

4.3 Empirical Process

This section discusses the empirical process that was followed to investigate the scalability of the MGPSO, PMGPSO, and other benchmark MaOO algorithms. Sections 4.3.1 through 4.3.4 respectively discuss the benchmark functions, the algorithms and parameter tuning approach, the performance measures, and the statistical analysis methodology.

4.3.1 Benchmark Functions

Seven problems from the DTLZ suite and nine problems from the WFG suite, as defined in sections 2.3.1 and 2.3.2, were used in this study. This collection of scalable benchmark functions present a mix of problems with different challenges and characteristics. These problems simulate challenging practical environments, making it suitable for testing algorithm scalability from MOO to MaOO. This thesis considers the benchmark problems with 3, 5, 8, 10, and

15 objectives with 30 decision variables, resulting in $(9 + 7) \times 5 = 80$ unique problems.

4.3.2 Algorithms and Parameter Tuning Approach

The experimental work for the original MGPSO algorithm, and all adaptations thereof, were conducted using CIlib[1] [24, 116] – a computational intelligence library written in Scala. The MGPSO algorithm and other versions thereof were implemented according to the coding standards of CIlib; that is, standards ensuring highly generic and correct code. CIlib aims to

- provide a type-safe library, preventing as many runtime errors and invalid data representations as possible;
- allow for the perfect reproduction of simulations, enabling researchers to validate and reuse previous work and published results with confidence; and
- enable composition, reducing the need to repeat implementations.

To ensure correctness and reliability, the implemented MGPSO algorithm and variations thereof were tested thoroughly throughout the implementation of the algorithm. Testing was mainly done in the form of property-based tests. Property-based[2] testing involves generating 100 random test cases to test different properties (functionality) of the algorithm.

The experimental work for the benchmark algorithms (i.e. the additional algorithms investigated throughout this study) was conducted using the PlatEMO[3] [153] framework.

For this part of the study, the following algorithms were investigated:

1. The multi-guide particle swarm optimization algorithm with the standard static archive balance coefficient update strategy (MGPSO$_{STD}$) [53, 136]. Refer to section 3.2.2.2 for detail.
2. The multi-guide particle swarm optimization algorithm with the random dynamic archive balance coefficient update strategy (MGPSO$_R$) [53, 136]. Refer to sections 3.2.2.2 and 3.17 for detail.

[1] Visit https://github.com/ciren/cilib and https://cilib.net/ for more information about CIlib. The source code can be found at https://github.com/CianSteenkamp96/CIlib_research/tree/v44.

[2] For more information regarding property-based testing, visit http://www.scalatest.org/user_guide/property_based_testing.

[3] Visit https://github.com/BIMK/PlatEMO for more information about PlatEMO. The source code can be found at https://github.com/CianSteenkamp96/PlatEMO_research/tree/master.

3. The multi-guide particle swarm optimization algorithm with the random per particle dynamic archive balance coefficient update strategy (MGPSO$_{RI}$) [53, 136]. Refer to sections 3.2.2.2 and 3.18 for detail.

4. The partial-dominance multi-guide particle swarm optimization algorithm with the standard static archive balance coefficient update strategy (PMGPSO$_{STD}$) [52, 53, 69]. Refer to sections 3.2.2.2 and 4.2 for detail.

5. The partial-dominance multi-guide particle swarm optimization algorithm with the random dynamic archive balance coefficient update strategy (PMGPSO$_{R}$) [52, 53, 69]. Refer to sections 3.17 and 4.2 for detail.

6. The partial-dominance multi-guide particle swarm optimization algorithm with the random per particle dynamic archive balance coefficient update strategy (PMGPSO$_{RI}$) [52, 53, 69]. Refer to sections 3.18 and 4.2 for detail.

7. CDAS-SMPSO [33]. Refer to section 3.2.2.1 for detail.

8. KnEA [174]. Refer to section 3.1.2.1 for detail.

9. MOEA/DD [95]. Refer to section 3.1.2.2 for detail.

10. NSGA-III [38]. Refer to section 3.1.2.3 for detail.

Each of the investigated algorithms has control parameters that need to be set which, to a large degree, determines the success of each algorithm. The process of finding an optimal control parameter configuration is referred to as parameter tuning. Parameter tuning is necessary because which configuration is best is very much problem-dependent. Finding an optimal control parameter configuration is a computationally expensive process due to the large volume of the control parameter problem space. Furthermore, *a priori* parameter tuning is a time-consuming procedure and assumes that the best parameters to employ do not change over time [67]. For the original PSO algorithm, results indicate that sampling control parameter values from a region known to contain promising control parameter configurations performed on par with the best control parameter configurations suggested in the literature [67]. Therefore, the MGPSO algorithm (and any adaptations thereof) used in this study, sampled new control parameter values from the convergent regions for each velocity update [20, 137]. Note that the convergent regions for the PMGPSO algorithm and MGPSO algorithm are the same since both algorithms use the same velocity update equation.

For each tunable algorithm (the CDAS-SMPSO algorithm, the KnEA, and the MOEA/DD), the optimal control parameter value combination for each problem was chosen as the setting with the best average IGD value over 30 independent runs. Each independent run consisted of 2000 iterations. Different parameter values were sampled for each parameter from the corresponding predefined parameter value domains. Note that the NSGA-III has no tunable parameters [38]. The general control parameter configurations used in this study are presented and discussed below.

For all 3-, 5-, 8-, 10-, and 15-objective problems, the number of candidate solutions (n_s) for each algorithm was set to 153, 126, 156, 110, and 135 respectively. For all relevant algorithms, the archive size was also set equal to n_s. These somewhat odd-looking values were chosen because the two-layered reference-point generation method [31], used by both the MOEA/DD [95] and NSGA-III [38], can only generate a certain number of points for each number of objectives. Note that the relatively small population size of 135 for 15 objectives is a consequence of the extremely high computational complexity required for the MOEA/DD and NSGA-III when solving MaOPs. That is, a larger population size would drastically increase the computational complexity, which would ultimately result in infeasible experimental runtimes [106]. Also, note that the same population sizes were used for each algorithm to ensure fair and unprejudiced comparisons, as recommended by [51].

Note that the initialization process of each algorithm was the same as described in Chapter 3, or as recommended or conventionally used by the original algorithms. A summary of the control parameter configurations for each of the considered algorithms is listed below.

MGPSO$_{STD}$, MGPSO$_{R}$, MGPSO$_{RI}$, PMGPSO$_{STD}$, PMGPSO$_{R}$, and PMGPSO$_{RI}$

The control parameter configurations for the different MGPSO and PMGPSO algorithms were as follows:

- The subswarm sizes of the MGPSO and PMGPSO algorithms, which are strictly speaking problem-dependent [20, 137], were set equal to the total swarm size divided by the number of objectives for convenience. That is, the size of each subswarm n_{s_1} to $n_{s_{n_m}}$ was (51, 51, 51) in the case of 3 objectives, (25, 25, 25, 25, 26) for 5 objectives, (19, 19, ..., 19, 20) for 8 objectives, (11, 11, ...) for 10 objectives, and (9, 9, ...) for 15 objectives.

- A tournament size of three was used for the archive guide selection process as recommended by [136].

- The inertia weight, w, and the acceleration coefficients c_1, c_2, and c_3 were resampled from the convergent regions [20, 137] at each iteration for each particle. In an effort to avoid sampling fruitlessly, default parameter values were assigned to the control parameters if no satisfactory values have been sampled after 10 tries. The default parameter values were w = 0.356, c_1 = 1.222, c_2 = 1.3, and c_3 = 1.517. These default values were calculated as the average optimal control parameter values for the

MGPSO algorithm on the three-objective WFG problems [136]. Note the default parameter values also satisfy the convergence/stability criteria.

- A maximum archive size equal to the total number of particles in all the subswarms combined was used.
- Velocity clamping was not used.
- The star neighbourhood topology was used.
- To ensure feasible guides, the personal and neighbourhood best position of each particle was only updated if the resulting position was feasible and better with reference to the corresponding objective.

CDAS-SMPSO

The control parameter configurations for the CDAS-SMPSO algorithm were as follows:

- At each iteration, w was randomly selected in the interval $[0, 0.8]$, r_1 and r_2 were randomly selected in $[0, 1]$, and c_1 and c_2 were randomly selected over the interval $[1.5, 2.5]$ [33]. The CDAS-SMPSO algorithm also guarantees convergent behaviour by ensuring that these randomly generated parameter values satisfy the necessary constraints of the constricted velocity [21, 23, 33, 112].
- The CDAS parameter, γ, was tuned in the range $\gamma \in \{0.25, 0.3, ..., 0.75\}$. The optimal γ-value for each of the problems investigated in this study is provided in table C.3.
- Polynomial mutation [36] with a distribution index of 20 was used.
- Polynomial mutation was applied at a probability of $\frac{1}{n}$ to 15% of the population, randomly selected [33, 106].

KnEA

The control parameter configurations for the KnEA were as follows:

- The rate of knee-points, κ, was tuned in the range $\kappa \in \{0.1, 0.2, ..., 0.9\}$. The optimal κ-value for each of the problems investigated in this study is provided in table C.1.
- The weighted distance metric employed by the KnEA used five-nearest neighbours [106, 174].
- Simulated binary crossover [2] with a distribution index of 30 and polynomial mutation [36] with a distribution index of 20 was used.
- The crossover probability was set to 1.0 and the mutation probability was set to $\frac{1}{n}$ [106, 174].

MOEA/DD
The control parameter configurations for the MOEA/DD were as follows:

- The PBI penalty parameter, φ, was set to 5.0 as recommended in [173].
- The neighbourhood size, ψ, was set equal to the number of weight vectors divided by 10, since it has been shown that neighbourhood sizes between 10 and 20 are generally sufficient [95]. The chosen population sizes all satisfy this condition.
- A neighbourhood selection probability, δ, of 0.9 is recommended by [94, 95], but this study tuned $\delta \in \{0.0, 0.1, ..., 1.0\}$. The optimal neighbourhood selection probability turned out to be different from this recommended value for most of the investigated problems. The tuned δ-value for each investigated problem is provided in table C.2.
- Simulated binary crossover [2] with a distribution index of 30 and polynomial mutation [36] with a distribution index of 20 was used [95, 106].
- The crossover probability was set to 1.0 and the mutation probability was set to $\frac{1}{n}$ [95, 106].

NSGA-III
The control parameter configurations were as follows:

- Simulated binary crossover [2] with a distribution index of 30 and polynomial mutation [36] with a distribution index of 20 was used [38, 106].
- The crossover probability was set to 1.0 and the mutation probability was set to $\frac{1}{n}$ [38, 106].

4.3.3 Performance Measures

This study quantified algorithm performance using the HV and IGD performance measures as defined in section 2.4. These quality indicators take into account both solution accuracy and solution diversity.

For all MaOPs investigated, this study approximated the HV using the Monte Carlo sampling technique [4]. This study used 10 000 000 sampling points for the Monte Carlo technique, as this number of sampling points have been shown to lead to 100% accuracy [4].

The IGD and HV performance measure values were calculated using the normalized objective function values (solutions) without outliers. Outlier removal is necessary to avoid anomalous solutions from skewing the results. Normalization is necessary to ensure a common scale in case of differing objective function value ranges.

The reference-point for all HV calculations was set to $\mathbf{1.1}$, as this vector has been shown to appropriately emphasize the convergence and diversity of the solution set [106]. For all IGD calculations, the true POSs were generated using the PlatEMO framework [153]. PlatEMO mathematically derives the true POFs by using the analytical forms of the benchmark suite functions. Each POS contained more than a 1000 Pareto-optimal decision vectors.

Both performance measures were calculated, using the final population or archive (normalized and without outliers), for each independent run for each algorithm on each problem. This study used 30 independent runs. Each independent run consisted of 2000 iterations/generations.

4.3.4 Statistical Analysis Process

After calculating the performance measure values, a series of non-parametric analysis of variance (ANOVA) statistical tests were executed. ANOVA tests are necessary because the average performance measure value for an algorithm on a problem may or may not be statistically significant, or different enough, from that of the other investigated algorithms. Claims about algorithm superiority can only be made if a statistically significant difference in algorithm performance exists. ANOVA tests are executed to determine if, and between which algorithms, statistically significant performance differences exist.

Non-parametric statistical tests are required due to the stochastic nature of the algorithms, i.e., the distribution of the solution samples cannot be assumed to be Gaussian. Some non-parametric outlier detection methods include Dbscan [54, 139] (which requires parameter tuning and training), isolation forests [100, 101] (an unsupervised clustering algorithm which also requires parameter tuning for each problem), the reference POF extreme values approach, and the interquartile range (IQR) [156, 183]. This study initially used the extreme values of the reference POFs for each problem to detect outliers. However, this resulted in 35% of the independent runs to be empty after outlier removal. With so little data left to analyze, no statistical inference could be made about algorithm superiority using this approach. The IQR outlier detection method removed all solutions for less than 1% of all independent runs. Therefore, this study used the intuitive and simple IQR outlier detection method to remove any outliers before normalizing all solutions to have values in the range $[0, 1]$.

This study used an extension/correction of the Friedman omnibus test [59, 60], referred to as the Iman-Davenport omnibus test [42, 78], at a confidence level of 95% to determine if a significant difference in algorithm performance existed between any of the algorithms. If a significant difference existed, two-tailed pairwise Wilcoxon signed rank sum tests [7, 43, 61, 72, 123, 169] at a confidence level of 95% were used to test for significant performance differences between specific algorithm pairs. To control the family-wise error rate (FWER) [143], the Holm [73] post-hoc p-value [143] adjustment was applied before accepting or rejecting the null hypothesis. Leaving the FWER uncontrolled can result in a cumulative error that significantly increases the probability of discovering undesired false positives (Type 1 errors [43, 61]). Note that this study appropriately makes use of two-tailed non-parametric paired tests, since the stochastic algorithms generated data (found solutions) for the same set of problems [43, 61, 123].

For each pair-wise test, if a statistically significant difference existed, the algorithm with the more desirable mean over 30 independent runs was given a win and the algorithm with the less desirable mean was given a loss. Note a low IGD average is desirable in contrast to a high HV average; that is, IGD should be minimized while HV should be maximized. The term difference is used to distinguish the performance of algorithms, which is simply the difference between pairwise wins and losses for a given algorithm. Additionally, the rank of each algorithm denotes the ranking in comparison to all other algorithms with respect to difference values for a given benchmark function [65, 106, 136]. The results in this study are reported as tables containing the overall wins, losses, difference, and rank across all the problems for each algorithm.

4.4 Results and Discussion

The results of the statistical analysis are shown in tables 4.1 to 4.20, which are then thoroughly analyzed and discussed. The tables contain the overall wins, losses, difference, and rank across the problems for each algorithm. Note that the top-three best overall ranks are highlighted for each table. The average, standard deviation, maximum, and minimum HV and IGD performance measure values are provided in Appendix D. Sections 4.4.1 and 4.4.2 discuss the findings with respect to HV and IGD respectively. Finally, section 4.4.3 makes some general remarks and summarizes the findings.

Table 4.1: HV Ranking for 3-objective DTLZ

Algorithm	**Result**	**3-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
MGPSO_{STD}	Wins	8	7	8	0	7	3	7	40
	Losses	0	2	0	3	2	6	0	13
	Difference	8	5	8	-3	5	-3	7	27
	Rank	1	3	1	6	3	7	1	1

Table 4.1: HV Ranking for 3-objective DTLZ (continue)

Algorithm	Result	3-objective DTLZ 1	2	3	4	5	6	7	Overall
$MGPSO_R$	Wins	4	8	4	0	8	4	7	35
	Losses	4	0	3	3	0	4	0	14
	Difference	0	8	1	-3	8	0	7	21
	Rank	5	1	4	6	1	5	1	3
$MGPSO_{RI}$	Wins	4	8	4	0	8	4	7	35
	Losses	4	0	3	4	0	4	0	15
	Difference	0	8	1	-4	8	0	7	20
	Rank	5	1	4	8	1	5	1	4
$PMGPSO_{STD}$	Wins	3	2	2	6	3	0	4	20
	Losses	6	6	7	0	3	7	3	32
	Difference	-3	-4	-5	6	0	-7	1	-12
	Rank	7	7	8	3	5	8	4	7
$PMGPSO_R$	Wins	1	0	0	7	1	0	4	13
	Losses	7	7	8	0	5	7	3	37
	Difference	-6	-7	-8	7	-4	-7	1	-24
	Rank	8	9	9	1	8	8	4	8
$PMGPSO_{RI}$	Wins	1	0	0	7	1	0	4	13
	Losses	7	8	8	0	6	7	3	39
	Difference	-6	-8	-8	7	-5	-7	1	-26
	Rank	8	10	9	1	9	8	4	9
CDAS-SMPSO	Wins	0	1	3	0	0	6	0	10
	Losses	9	6	6	5	8	2	9	45
	Difference	-9	-5	-3	-5	-8	4	-9	-35
	Rank	10	8	7	9	10	3	10	10

Table 4.1: HV Ranking for 3-objective DTLZ (continue)

Algorithm	**Result**	**3-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
KnEA	Wins	8	6	8	3	0	9	2	36
	Losses	0	3	0	2	3	0	6	14
	Difference	8	3	8	1	-3	9	-4	22
	Rank	1	4	1	4	7	1	7	2
MOEA/DD	Wins	6	5	4	2	4	8	2	31
	Losses	2	4	3	3	3	1	6	22
	Difference	4	1	1	-1	1	7	-4	9
	Rank	3	5	4	5	4	2	7	5
NSGA-III	Wins	6	4	7	0	2	6	1	26
	Losses	2	5	2	5	4	2	8	28
	Difference	4	-1	5	-5	-2	4	-7	-2
	Rank	3	6	3	9	6	3	9	6

Table 4.2: HV Ranking for 5-objective DTLZ

Algorithm	**Result**	**5-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
MGPSO_{STD}	Wins	6	4	6	4	7	2	7	36
	Losses	2	1	2	3	0	2	0	10
	Difference	4	3	4	1	7	0	7	26
	Rank	3	3	3	6	1	5	1	3
MGPSO_{R}	Wins	5	4	4	4	7	4	7	35
	Losses	3	1	3	1	0	2	0	10
	Difference	2	3	1	3	7	2	7	25
	Rank	5	3	4	4	1	3	1	4

Table 4.2: HV Ranking for 5-objective DTLZ (continue)

Algorithm	**Result**	**5-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$MGPSO_{RI}$	Wins	5	5	4	5	7	4	7	37
	Losses	2	1	3	1	0	2	0	9
	Difference	3	4	1	4	7	2	7	28
	Rank	4	2	4	3	1	3	1	2
$PMGPSO_{STD}$	Wins	2	0	1	2	2	1	4	12
	Losses	5	6	5	6	4	5	3	34
	Difference	-3	-6	-4	-4	-2	-4	1	-22
	Rank	6	8	7	7	5	9	4	6
$PMGPSO_{R}$	Wins	2	0	0	0	2	1	4	9
	Losses	5	6	7	7	4	4	3	36
	Difference	-3	-6	-7	-7	-2	-3	1	-27
	Rank	6	8	10	9	5	7	4	8
$PMGPSO_{RI}$	Wins	2	0	0	0	2	1	4	9
	Losses	5	6	6	7	4	4	3	35
	Difference	-3	-6	-6	-7	-2	-3	1	-26
	Rank	6	8	9	9	5	7	4	7
CDAS-SMPSO	Wins	1	0	0	4	1	1	0	7
	Losses	8	5	5	2	7	2	9	38
	Difference	-7	-5	-5	2	-6	-1	-9	-31
	Rank	9	7	8	5	9	6	10	9
KnEA	Wins	0	3	2	0	0	0	2	7
	Losses	9	2	2	6	9	9	6	43
	Difference	-9	1	0	-6	-9	-9	-4	-35
	Rank	10	6	6	8	10	10	7	10

Table 4.2: HV Ranking for 5-objective DTLZ (continue)

Algorithm	**Result**	**5-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
MOEA/DD	Wins	8	9	8	9	6	9	1	50
	Losses	0	0	1	0	3	0	6	10
	Difference	8	9	7	9	3	9	-5	40
	Rank	1	1	2	1	4	1	8	1
NSGA-III	Wins	8	4	9	6	1	8	1	37
	Losses	0	1	0	1	4	1	7	14
	Difference	8	3	9	5	-3	7	-6	23
	Rank	1	3	1	2	8	2	9	5

Table 4.3: HV Ranking for 8-objective DTLZ

Algorithm	**Result**	**8-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$MGPSO_{STD}$	Wins	6	6	5	6	7	1	8	39
	Losses	3	0	2	3	0	3	0	11
	Difference	3	6	3	3	7	-2	8	28
	Rank	4	1	3	4	1	4	1	1
$MGPSO_{R}$	Wins	4	6	4	3	7	1	4	29
	Losses	4	0	3	4	0	3	1	15
	Difference	0	6	1	-1	7	-2	3	14
	Rank	5	1	5	5	1	4	4	5
$MGPSO_{RI}$	Wins	4	6	4	2	7	1	5	29
	Losses	4	0	2	4	0	3	1	14
	Difference	0	6	2	-2	7	-2	4	15
	Rank	5	1	4	7	1	4	2	4

Table 4.3: HV Ranking for 8-objective DTLZ (continue)

Algorithm	**Result**	**8-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
PMGPSO_{STD}	Wins	1	1	0	3	6	1	4	16
	Losses	6	5	7	4	3	3	0	28
	Difference	-5	-4	-7	-1	3	-2	4	-12
	Rank	7	6	8	5	4	4	2	6
PMGPSO_{R}	Wins	1	1	0	1	4	1	4	12
	Losses	6	5	7	6	4	3	1	32
	Difference	-5	-4	-7	-5	0	-2	3	-20
	Rank	7	6	8	8	5	4	4	8
PMGPSO_{RI}	Wins	1	1	0	0	4	1	4	11
	Losses	6	5	7	7	4	3	2	34
	Difference	-5	-4	-7	-7	0	-2	2	-23
	Rank	7	6	8	9	5	4	6	9
CDAS-SMPSO	Wins	0	1	3	7	1	7	0	19
	Losses	9	5	5	2	6	1	9	37
	Difference	-9	-4	-2	5	-5	6	-9	-18
	Rank	10	6	7	3	7	2	10	7
KnEA	Wins	9	0	3	0	0	0	2	14
	Losses	0	9	2	8	9	9	6	43
	Difference	9	-9	1	-8	-9	-9	-4	-29
	Rank	1	10	5	10	10	10	7	10
MOEA/DD	Wins	7	6	8	9	1	9	1	41
	Losses	1	0	0	0	6	0	8	15
	Difference	6	6	8	9	-5	9	-7	26
	Rank	2	1	1	1	7	1	9	2

Table 4.3: HV Ranking for 8-objective DTLZ (continue)

Algorithm	Result	8-objective DTLZ 1	2	3	4	5	6	7	Overall
NSGA-III	Wins	7	5	8	8	1	7	2	38
	Losses	1	4	0	1	6	1	6	19
	Difference	6	1	8	7	-5	6	-4	19
	Rank	2	5	1	2	7	2	7	3

Table 4.4: HV Ranking for 10-objective DTLZ

Algorithm	Result	10-objective DTLZ 1	2	3	4	5	6	7	Overall
$MGPSO_{STD}$	Wins	5	7	3	5	7	2	8	37
	Losses	1	0	2	2	0	1	0	6
	Difference	4	7	1	3	7	1	8	31
	Rank	2	2	5	3	1	3	1	1
$MGPSO_{R}$	Wins	5	7	3	0	7	2	4	28
	Losses	1	1	2	4	0	2	2	12
	Difference	4	6	1	-4	7	0	2	16
	Rank	2	3	5	8	1	7	3	3
$MGPSO_{RI}$	Wins	5	8	3	0	7	2	4	29
	Losses	1	0	2	3	0	1	2	9
	Difference	4	8	1	-3	7	1	2	20
	Rank	2	1	5	7	1	3	3	2
$PMGPSO_{STD}$	Wins	1	1	0	3	4	2	8	19
	Losses	5	3	7	2	3	1	0	21
	Difference	-4	-2	-7	1	1	1	8	-2
	Rank	7	5	8	4	4	3	1	5

Table 4.4: HV Ranking for 10-objective DTLZ (continue)

Algorithm	**Result**	**10-objective DTLZ 1**	**2**	**3**	**4**	**5**	**6**	**7**	**Overall**
PMGPSO_R	Wins	1	1	0	0	4	3	4	13
	Losses	5	4	7	5	3	1	2	27
	Difference	-4	-3	-7	-5	1	2	2	-14
	Rank	7	8	8	9	4	2	3	8
PMGPSO_{RI}	Wins	1	1	0	0	4	2	4	12
	Losses	5	4	7	5	3	1	2	27
	Difference	-4	-3	-7	-5	1	1	2	-15
	Rank	7	8	8	9	4	3	3	9
CDAS-SMPSO	Wins	0	3	3	0	3	9	0	18
	Losses	9	3	1	0	6	0	9	28
	Difference	-9	0	2	0	-3	9	-9	-10
	Rank	10	4	3	5	7	1	10	6
KnEA	Wins	9	1	3	2	1	0	2	18
	Losses	0	3	1	3	7	8	7	29
	Difference	9	-2	2	-1	-6	-8	-5	-11
	Rank	1	5	3	6	8	10	8	7
MOEA/DD	Wins	1	0	6	7	1	0	1	16
	Losses	4	9	0	0	7	7	8	35
	Difference	-3	-9	6	7	-6	-7	-7	-19
	Rank	6	10	2	1	8	9	9	10
NSGA-III	Wins	4	1	8	7	0	1	3	24
	Losses	1	3	0	0	9	1	6	20
	Difference	3	-2	8	7	-9	0	-3	4
	Rank	5	5	1	1	10	7	7	4

Table 4.5: HV Ranking for 15-objective DTLZ

Algorithm	**Result**	**15-objective DTLZ 1**	**2**	**3**	**4**	**5**	**6**	**7**	**Overall**
$MGPSO_{STD}$	Wins	5	7	4	6	7	1	8	38
	Losses	0	0	1	0	0	1	0	2
	Difference	5	7	3	6	7	0	8	36
	Rank	2	1	4	1	1	6	1	1
$MGPSO_{R}$	Wins	5	7	4	0	7	2	4	29
	Losses	1	0	2	2	0	1	2	8
	Difference	4	7	2	-2	7	1	2	21
	Rank	4	1	5	6	1	2	3	2
$MGPSO_{RI}$	Wins	5	7	4	0	7	0	4	27
	Losses	1	0	2	2	0	1	2	8
	Difference	4	7	2	-2	7	-1	2	19
	Rank	4	1	5	6	1	8	3	3
$PMGPSO_{STD}$	Wins	1	3	0	5	4	2	8	23
	Losses	5	3	7	1	3	1	0	20
	Difference	-4	0	-7	4	1	1	8	3
	Rank	6	4	8	2	4	2	1	4
$PMGPSO_{R}$	Wins	1	3	0	0	4	2	4	14
	Losses	5	3	7	2	3	1	2	23
	Difference	-4	0	-7	-2	1	1	2	-9
	Rank	6	4	8	6	4	2	3	5
$PMGPSO_{RI}$	Wins	1	3	0	0	4	2	4	14
	Losses	5	3	7	2	3	1	2	23
	Difference	-4	0	-7	-2	1	1	2	-9
	Rank	6	4	8	6	4	2	3	5

Table 4.5: HV Ranking for 15-objective DTLZ (continue)

Algorithm	Result	15-objective DTLZ 1	2	3	4	5	6	7	Overall
CDAS-SMPSO	Wins	0	2	3	0	3	8	0	16
	Losses	9	3	6	0	6	0	9	33
	Difference	-9	-1	-3	0	-3	8	-9	-17
	Rank	10	7	7	3	7	1	10	9
KnEA	Wins	7	1	4	0	0	0	2	14
	Losses	0	7	0	2	7	6	7	29
	Difference	7	-6	4	-2	-7	-6	-5	-15
	Rank	1	9	3	6	8	10	8	8
MOEA/DD	Wins	5	0	7	0	0	0	1	13
	Losses	0	9	0	0	7	0	8	24
	Difference	5	-9	7	0	-7	0	-7	-11
	Rank	2	10	1	3	8	6	9	7
NSGA-III	Wins	1	1	6	0	0	0	3	11
	Losses	5	6	0	0	7	5	6	29
	Difference	-4	-5	6	0	-7	-5	-3	-18
	Rank	6	8	2	3	8	9	7	10

Table 4.6: HV Ranking for 3-objective WFG

Algorithm	Result	3-objective WFG 1	2	3	4	5	6	7	8	9	Overall
$MGPSO_{STD}$	Wins	3	3	7	5	3	5	2	3	7	38
	Losses	6	4	2	1	1	3	4	5	1	27
	Difference	-3	-1	5	4	2	2	-2	-2	6	11
	Rank	7	7	3	2	4	4	5	7	3	5

Table 4.6: HV Ranking for 3-objective WFG (continue)

Algorithm	**Result**	**3-objective WFG 1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Overall**
$MGPSO_R$	Wins	4	3	8	5	3	7	2	4	7	43
	Losses	4	3	0	1	1	1	4	4	0	18
	Difference	0	0	8	4	2	6	-2	0	7	25
	Rank	5	5	1	2	4	2	5	5	2	4
$MGPSO_{RI}$	Wins	4	4	8	5	3	7	2	3	8	44
	Losses	4	3	0	1	1	1	4	4	0	18
	Difference	0	1	8	4	2	6	-2	-1	8	26
	Rank	5	4	1	2	4	2	5	6	1	3
$PMGPSO_{STD}$	Wins	0	0	1	1	0	0	2	0	3	7
	Losses	7	7	5	6	7	7	4	7	3	53
	Difference	-7	-7	-4	-5	-7	-7	-2	-7	0	-46
	Rank	8	8	6	9	8	8	5	8	6	8
$PMGPSO_R$	Wins	0	0	1	1	0	0	0	0	4	6
	Losses	7	7	5	5	7	7	8	7	3	56
	Difference	-7	-7	-4	-4	-7	-7	-8	-7	1	-50
	Rank	8	8	6	7	8	8	9	8	4	9
$PMGPSO_{RI}$	Wins	0	0	1	1	0	0	0	0	4	6
	Losses	7	7	5	5	7	7	8	7	3	56
	Difference	-7	-7	-4	-4	-7	-7	-8	-7	1	-50
	Rank	8	8	6	7	8	8	9	8	4	9
CDAS-SMPSO	Wins	7	8	6	5	4	5	8	8	3	54
	Losses	0	0	3	1	1	3	1	1	5	15
	Difference	7	8	3	4	3	2	7	7	-2	39
	Rank	2	1	4	2	2	4	2	2	7	2

Table 4.6: HV Ranking for 3-objective WFG (continue)

Algorithm	**Result**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Overall**
KnEA	Wins	6	8	1	9	9	9	9	9	2	62
	Losses	3	0	5	0	0	0	0	0	7	15
	Difference	3	8	-4	9	9	9	9	9	-5	47
	Rank	4	1	6	1	1	1	1	1	8	1
MOEA/DD	Wins	8	7	0	0	3	3	6	6	0	33
	Losses	0	2	9	9	3	5	3	3	8	42
	Difference	8	5	-9	-9	0	-2	3	3	-8	-9
	Rank	1	3	10	10	7	6	4	4	9	7
NSGA-III	Wins	7	3	5	2	4	3	7	7	0	38
	Losses	1	3	4	5	1	5	2	2	8	31
	Difference	6	0	1	-3	3	-2	5	5	-8	7
	Rank	3	5	5	6	2	6	3	3	9	6

Table 4.7: HV Ranking for 5-objective WFG

Algorithm	**Result**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Overall**
MGPSO_{STD}	Wins	3	5	7	2	1	3	2	3	7	33
	Losses	6	1	2	4	4	6	4	4	0	31
	Difference	-3	4	5	-2	-3	-3	-2	-1	7	2
	Rank	7	2	3	7	6	7	5	5	1	7
MGPSO_{R}	Wins	4	5	9	3	2	4	2	3	7	39
	Losses	4	1	0	4	4	3	4	4	0	24
	Difference	0	4	9	-1	-2	1	-2	-1	7	15
	Rank	5	2	1	5	5	5	5	5	1	3

Table 4.7: HV Ranking for 5-objective WFG (continue)

Algorithm	**Result**	**5-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
$MGPSO_{RI}$	Wins	4	5	8	3	1	4	2	3	7	37
	Losses	4	1	1	4	4	3	4	4	0	25
	Difference	0	4	7	-1	-3	1	-2	-1	7	12
	Rank	5	2	2	5	6	5	5	5	1	5
$PMGPSO_{STD}$	Wins	0	1	6	0	0	0	2	0	3	12
	Losses	7	5	3	7	4	7	4	7	3	47
	Difference	-7	-4	3	-7	-4	-7	-2	-7	0	-35
	Rank	8	6	4	9	8	8	5	8	4	8
$PMGPSO_{R}$	Wins	0	1	4	0	0	0	0	0	3	8
	Losses	7	5	4	7	7	7	8	7	3	55
	Difference	-7	-4	0	-7	-7	-7	-8	-7	0	-47
	Rank	8	6	5	9	10	8	9	8	4	10
$PMGPSO_{RI}$	Wins	0	1	4	0	0	0	0	0	3	8
	Losses	7	5	4	6	5	7	8	7	3	52
	Difference	-7	-4	0	-6	-5	-7	-8	-7	0	-44
	Rank	8	6	5	8	9	8	9	8	4	9
CDAS-SMPSO	Wins	7	8	3	7	6	4	6	6	2	49
	Losses	0	0	6	1	3	2	1	3	6	22
	Difference	7	8	-3	6	3	2	5	3	-4	27
	Rank	1	1	7	2	4	4	2	4	8	2
KnEA	Wins	6	4	1	9	9	9	9	9	2	58
	Losses	3	0	8	0	0	0	0	0	3	14
	Difference	3	4	-7	9	9	9	9	9	-1	44
	Rank	4	2	9	1	1	1	1	1	7	1

Table 4.7: HV Ranking for 5-objective WFG (continue)

Algorithm	Result	1	2	3	4	5	6	7	8	9	Overall
MOEA/DD	Wins	7	0	0	7	7	7	6	7	0	41
	Losses	0	8	9	1	1	1	1	1	8	30
	Difference	7	-8	-9	6	6	6	5	6	-8	11
	Rank	1	10	10	2	2	2	2	2	9	6
NSGA-III	Wins	7	0	2	6	7	6	6	7	0	41
	Losses	0	4	7	3	1	1	1	1	8	26
	Difference	7	-4	-5	3	6	5	5	6	-8	15
	Rank	1	6	8	4	2	3	2	2	9	3

Table 4.8: HV Ranking for 8-objective WFG

Algorithm	Result	1	2	3	4	5	6	7	8	9	Overall
MGPSO_{STD}	Wins	4	7	7	3	4	5	1	3	7	41
	Losses	2	0	0	0	2	0	2	3	0	9
	Difference	2	7	7	3	2	5	-1	0	7	32
	Rank	3	1	1	2	4	3	4	4	1	3
MGPSO_{R}	Wins	8	7	7	3	1	7	1	3	7	44
	Losses	0	0	0	0	5	0	3	3	0	11
	Difference	8	7	7	3	-4	7	-2	0	7	33
	Rank	1	1	1	2	7	1	5	4	1	2
MGPSO_{RI}	Wins	8	7	7	5	1	7	1	3	7	46
	Losses	0	0	0	0	5	0	3	3	0	11
	Difference	8	7	7	5	-4	7	-2	0	7	35
	Rank	1	1	1	1	7	1	5	4	1	1

Table 4.8: HV Ranking for 8-objective WFG (continue)

Algorithm	Result	8-objective WFG 1	2	3	4	5	6	7	8	9	Overall
$PMGPSO_{STD}$	Wins	0	6	5	3	4	4	1	1	4	28
	Losses	6	3	3	1	2	2	3	3	3	26
	Difference	-6	3	2	2	2	2	-2	-2	1	2
	Rank	8	4	4	6	4	4	5	7	4	5
$PMGPSO_{R}$	Wins	0	4	4	3	1	4	1	1	4	22
	Losses	6	4	4	0	5	3	3	6	3	34
	Difference	-6	0	0	3	-4	1	-2	-5	1	-12
	Rank	8	5	6	2	7	6	5	8	4	8
$PMGPSO_{RI}$	Wins	0	4	4	3	1	4	1	1	4	22
	Losses	6	4	3	1	3	2	3	6	3	31
	Difference	-6	0	1	2	-2	2	-2	-5	1	-9
	Rank	8	5	5	6	6	4	5	8	4	7
CDAS-SMPSO	Wins	0	1	3	3	9	1	8	7	2	34
	Losses	3	6	6	0	0	6	0	0	6	27
	Difference	-3	-5	-3	3	9	-5	8	7	-4	7
	Rank	7	7	7	2	1	7	1	2	7	4
KnEA	Wins	3	1	0	0	0	0	0	0	0	4
	Losses	2	6	8	9	9	9	9	9	9	70
	Difference	1	-5	-8	-9	-9	-9	-9	-9	-9	-66
	Rank	4	7	9	10	10	10	10	10	10	10
MOEA/DD	Wins	3	0	0	1	5	1	6	7	1	24
	Losses	2	9	8	7	1	6	2	1	8	44
	Difference	1	-9	-8	-6	4	-5	4	6	-7	-20
	Rank	4	10	9	8	3	7	3	3	9	9

Table 4.8: HV Ranking for 8-objective WFG (continue)

Algorithm	**Result**	**8-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
NSGA-III	Wins	3	1	2	1	7	1	8	8	2	33
	Losses	2	6	7	7	1	6	0	0	6	35
	Difference	1	-5	-5	-6	6	-5	8	8	-4	-2
	Rank	4	7	8	8	2	7	1	1	7	6

Table 4.9: HV Ranking for 10-objective WFG

Algorithm	**Result**	**10-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
MGPSO_{STD}	Wins	4	7	7	4	7	4	2	8	5	48
	Losses	2	0	0	1	0	1	0	0	0	4
	Difference	2	7	7	3	7	3	2	8	5	44
	Rank	3	1	1	4	1	3	1	1	3	2
MGPSO_{R}	Wins	8	7	7	5	2	4	2	6	6	47
	Losses	0	0	0	0	2	0	0	1	0	3
	Difference	8	7	7	5	0	4	2	5	6	44
	Rank	1	1	1	2	4	2	1	3	2	2
MGPSO_{RI}	Wins	8	7	7	7	2	8	2	3	7	51
	Losses	0	0	0	0	2	0	0	2	0	4
	Difference	8	7	7	7	0	8	2	1	7	47
	Rank	1	1	1	1	4	1	1	4	1	1
PMGPSO_{STD}	Wins	0	4	4	3	7	4	2	7	4	35
	Losses	6	3	3	2	0	1	0	0	2	17
	Difference	-6	1	1	1	7	3	2	7	2	18
	Rank	9	4	4	6	1	3	1	2	5	4

Table 4.9: HV Ranking for 10-objective WFG (continue)

Algorithm	Result	10-objective WFG 1	2	3	4	5	6	7	8	9	Overall
$PMGPSO_R$	Wins	1	4	4	4	2	4	2	3	4	28
	Losses	5	3	3	0	2	1	0	3	1	18
	Difference	-4	1	1	4	0	3	2	0	3	10
	Rank	7	4	4	3	4	3	1	5	4	5
$PMGPSO_{RI}$	Wins	0	4	4	4	2	4	2	3	4	27
	Losses	5	3	3	1	2	1	0	3	3	21
	Difference	-5	1	1	3	0	3	2	0	1	6
	Rank	8	4	4	4	4	3	1	5	6	6
CDAS-SMPSO	Wins	0	1	3	3	2	2	2	2	0	15
	Losses	6	8	6	5	2	6	0	3	6	42
	Difference	-6	-7	-3	-2	0	-4	2	-1	-6	-27
	Rank	9	9	7	7	4	7	1	7	7	7
KnEA	Wins	4	2	1	0	2	2	0	0	0	11
	Losses	2	6	8	7	0	6	8	8	6	51
	Difference	2	-4	-7	-7	2	-4	-8	-8	-6	-40
	Rank	3	7	9	8	3	7	9	9	7	9
MOEA/DD	Wins	4	0	0	0	0	0	0	0	0	4
	Losses	2	9	9	7	9	9	8	8	6	67
	Difference	2	-9	-9	-7	-9	-9	-8	-8	-6	-63
	Rank	3	10	10	8	10	10	9	9	7	10
NSGA-III	Wins	1	2	2	0	1	1	2	2	0	11
	Losses	2	6	7	7	8	8	0	6	6	50
	Difference	-1	-4	-5	-7	-7	-7	2	-4	-6	-39
	Rank	6	7	8	8	9	9	1	8	7	8

Table 4.10: HV Ranking for 15-objective WFG

Algorithm	Result	15-objective WFG									Overall
		1	2	3	4	5	6	7	8	9	
$MGPSO_{STD}$	Wins	4	6	7	4	4	3	1	1	2	32
	Losses	0	1	0	0	1	2	2	1	0	7
	Difference	4	5	7	4	3	1	-1	0	2	25
	Rank	1	2	1	3	3	5	4	3	2	3
$MGPSO_{R}$	Wins	4	6	7	6	2	6	1	1	3	36
	Losses	0	1	0	0	5	1	2	1	0	10
	Difference	4	5	7	6	-3	5	-1	0	3	26
	Rank	1	2	1	1	7	2	4	3	1	2
$MGPSO_{RI}$	Wins	4	6	7	6	2	4	1	1	2	33
	Losses	0	1	0	0	5	1	2	1	0	10
	Difference	4	5	7	6	-3	3	-1	0	2	23
	Rank	1	2	1	1	7	3	4	3	2	4
$PMGPSO_{STD}$	Wins	0	2	4	2	6	3	1	1	2	21
	Losses	3	4	3	3	1	3	2	1	0	20
	Difference	-3	-2	1	-1	5	0	-1	0	2	1
	Rank	7	5	4	7	2	7	4	3	2	5
$PMGPSO_{R}$	Wins	0	2	4	2	4	3	1	1	2	19
	Losses	3	4	3	3	2	2	2	1	0	20
	Difference	-3	-2	1	-1	2	1	-1	0	2	-1
	Rank	7	5	4	7	4	5	4	3	2	7
$PMGPSO_{RI}$	Wins	0	2	4	2	4	3	1	1	2	19
	Losses	3	4	3	2	2	1	2	1	0	18
	Difference	-3	-2	1	0	2	2	-1	0	2	1
	Rank	7	5	4	5	4	4	4	3	2	5

Table 4.10: HV Ranking for 15-objective WFG (continue)

Algorithm	Result	15-objective WFG									Overall
		1	2	3	4	5	6	7	8	9	
CDAS-SMPSO	Wins	0	0	3	1	0	1	1	1	1	8
	Losses	0	7	6	8	8	7	1	1	8	46
	Difference	0	-7	-3	-7	-8	-6	0	0	-7	-38
	Rank	4	9	7	9	9	8	3	3	9	9
KnEA	Wins	0	1	0	0	0	0	0	0	0	1
	Losses	3	4	8	9	8	9	9	9	9	68
	Difference	-3	-3	-8	-9	-8	-9	-9	-9	-9	-67
	Rank	7	8	9	10	9	10	10	10	10	10
MOEA/DD	Wins	0	0	0	2	2	1	7	1	2	15
	Losses	0	8	8	0	1	7	0	0	1	25
	Difference	0	-8	-8	2	1	-6	7	1	1	-10
	Rank	4	10	9	4	6	8	2	2	8	8
NSGA-III	Wins	0	9	2	2	9	9	8	8	2	49
	Losses	0	0	7	2	0	0	0	0	0	9
	Difference	0	9	-5	0	9	9	8	8	2	40
	Rank	4	1	8	5	1	1	1	1	2	1

Table 4.11: IGD Ranking for 3-objective DTLZ

Algorithm	Result	3-objective DTLZ							Overall
		1	2	3	4	5	6	7	
$MGPSO_{STD}$	Wins	1	2	2	0	0	4	2	11
	Losses	4	6	6	7	6	3	7	39
	Difference	-3	-4	-4	-7	-6	1	-5	-28
	Rank	6	7	7	8	8	4	8	9

Table 4.11: IGD Ranking for 3-objective DTLZ (continue)

Algorithm	Result	3-objective DTLZ 1	2	3	4	5	6	7	Overall
$MGPSO_R$	Wins	8	0	5	0	0	4	0	17
	Losses	0	8	3	7	7	3	8	36
	Difference	8	-8	2	-7	-7	1	-8	-19
	Rank	1	9	4	8	10	4	9	8
$MGPSO_{RI}$	Wins	8	0	5	0	0	4	0	17
	Losses	0	8	3	7	6	3	8	35
	Difference	8	-8	2	-7	-6	1	-8	-18
	Rank	1	9	4	8	8	4	9	7
$PMGPSO_{STD}$	Wins	1	7	7	6	7	7	7	42
	Losses	5	0	0	2	0	0	0	7
	Difference	-4	7	7	4	7	7	7	35
	Rank	7	1	1	4	1	1	1	3
$PMGPSO_R$	Wins	1	7	7	7	7	7	7	43
	Losses	5	0	0	0	0	0	0	5
	Difference	-4	7	7	7	7	7	7	38
	Rank	7	1	1	1	1	1	1	1
$PMGPSO_{RI}$	Wins	1	7	7	6	7	7	7	42
	Losses	5	0	0	0	0	0	0	5
	Difference	-4	7	7	6	7	7	7	37
	Rank	7	1	1	3	1	1	1	2
CDAS-SMPSO	Wins	0	2	0	3	1	0	3	9
	Losses	9	6	9	6	6	9	6	51
	Difference	-9	-4	-9	-3	-5	-9	-3	-42
	Rank	10	7	10	7	7	10	7	10

Table 4.11: IGD Ranking for 3-objective DTLZ (continue)

Algorithm	Result	3-objective DTLZ 1	2	3	4	5	6	7	Overall
KnEA	Wins	4	4	1	4	4	1	5	23
	Losses	4	5	8	4	5	8	4	38
	Difference	0	-1	-7	0	-1	-7	1	-15
	Rank	5	6	9	5	6	9	5	6
MOEA/DD	Wins	7	6	4	7	6	3	4	37
	Losses	2	3	5	0	3	6	5	24
	Difference	5	3	-1	7	3	-3	-1	13
	Rank	3	4	6	1	4	7	6	4
NSGA-III	Wins	6	5	2	4	5	2	6	30
	Losses	3	4	6	4	4	7	3	31
	Difference	3	1	-4	0	1	-5	3	-1
	Rank	4	5	7	5	5	8	4	5

Table 4.12: IGD Ranking for 5-objective DTLZ

Algorithm	Result	5-objective DTLZ 1	2	3	4	5	6	7	Overall
MGPSO_{STD}	Wins	6	3	4	0	2	4	0	19
	Losses	2	5	5	3	5	0	6	26
	Difference	4	-2	-1	-3	-3	4	-6	-7
	Rank	3	6	6	9	6	1	9	8
MGPSO_{R}	Wins	5	2	5	0	2	4	1	19
	Losses	3	5	3	2	5	0	5	23
	Difference	2	-3	2	-2	-3	4	-4	-4
	Rank	5	7	4	4	6	1	7	6

Table 4.12: IGD Ranking for 5-objective DTLZ (continue)

Algorithm	**Result**	**5-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$MGPSO_{RI}$	Wins	5	2	5	0	2	4	0	18
	Losses	2	6	3	2	5	0	5	23
	Difference	3	-4	2	-2	-3	4	-5	-5
	Rank	4	8	4	4	6	1	8	7
$PMGPSO_{STD}$	Wins	1	7	7	0	7	4	7	33
	Losses	6	0	0	2	0	0	0	8
	Difference	-5	7	7	-2	7	4	7	25
	Rank	7	1	1	4	1	1	1	1
$PMGPSO_{R}$	Wins	1	7	7	0	7	4	7	33
	Losses	6	0	0	3	0	0	0	9
	Difference	-5	7	7	-3	7	4	7	24
	Rank	7	1	1	9	1	1	1	3
$PMGPSO_{RI}$	Wins	1	7	7	0	7	4	7	33
	Losses	6	0	0	2	0	0	0	8
	Difference	-5	7	7	-2	7	4	7	25
	Rank	7	1	1	4	1	1	1	1
CDAS-SMPSO	Wins	4	0	0	2	1	0	0	7
	Losses	5	9	9	2	8	8	7	48
	Difference	-1	-9	-9	0	-7	-8	-7	-41
	Rank	6	10	10	3	9	9	10	10
KnEA	Wins	0	1	1	0	0	0	5	7
	Losses	9	8	8	2	9	8	4	48
	Difference	-9	-7	-7	-2	-9	-8	1	-41
	Rank	10	9	9	4	10	9	5	10

Table 4.12: IGD Ranking for 5-objective DTLZ (continue)

Algorithm	Result	5-objective DTLZ 1	2	3	4	5	6	7	Overall
MOEA/DD	Wins	9	5	2	8	5	3	2	34
	Losses	0	4	7	0	4	6	5	26
	Difference	9	1	-5	8	1	-3	-3	8
	Rank	1	5	8	1	5	7	6	5
NSGA-III	Wins	8	6	3	8	6	2	6	39
	Losses	1	3	6	0	3	7	3	23
	Difference	7	3	-3	8	3	-5	3	16
	Rank	2	4	7	1	4	8	4	4

Table 4.13: IGD Ranking for 8-objective DTLZ

Algorithm	Result	8-objective DTLZ 1	2	3	4	5	6	7	Overall
MGPSO_{STD}	Wins	3	2	4	6	2	4	2	23
	Losses	4	3	3	2	5	0	7	24
	Difference	-1	-1	1	4	-3	4	-5	-1
	Rank	5	4	4	3	6	1	8	6
MGPSO_{R}	Wins	3	2	4	1	2	4	3	19
	Losses	4	3	3	4	5	0	4	23
	Difference	-1	-1	1	-3	-3	4	-1	-4
	Rank	5	4	4	6	6	1	5	7
MGPSO_{RI}	Wins	3	2	4	1	2	4	3	19
	Losses	4	3	3	4	5	0	4	23
	Difference	-1	-1	1	-3	-3	4	-1	-4
	Rank	5	4	4	6	6	1	5	7

Table 4.13: IGD Ranking for 8-objective DTLZ (continue)

Algorithm	**Result**	**8-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
PMGPSO_{STD}	Wins	0	7	7	2	7	4	6	33
	Losses	7	0	0	4	0	0	1	12
	Difference	-7	7	7	-2	7	4	5	21
	Rank	8	1	1	5	1	1	2	1
PMGPSO_{R}	Wins	0	7	7	1	7	4	6	32
	Losses	7	0	0	4	0	0	1	12
	Difference	-7	7	7	-3	7	4	5	20
	Rank	8	1	1	6	1	1	2	2
PMGPSO_{RI}	Wins	0	7	7	1	7	4	6	32
	Losses	7	0	0	5	0	0	1	13
	Difference	-7	7	7	-4	7	4	5	19
	Rank	8	1	1	9	1	1	2	3
CDAS-SMPSO	Wins	6	1	0	6	1	0	0	14
	Losses	2	8	9	2	8	8	9	46
	Difference	4	-7	-9	4	-7	-8	-9	-32
	Rank	3	9	10	3	9	9	10	9
KnEA	Wins	8	0	1	0	0	0	3	12
	Losses	1	9	8	9	9	8	4	48
	Difference	7	-9	-7	-9	-9	-8	-1	-36
	Rank	2	10	9	10	10	9	5	10
MOEA/DD	Wins	9	2	2	8	5	2	1	29
	Losses	0	3	6	0	4	6	8	27
	Difference	9	-1	-4	8	1	-4	-7	2
	Rank	1	4	7	1	5	7	9	5

Table 4.13: IGD Ranking for 8-objective DTLZ (continue)

Algorithm	**Result**	**8-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
NSGA-III	Wins	6	2	2	8	6	2	9	35
	Losses	2	3	6	0	3	6	0	20
	Difference	4	-1	-4	8	3	-4	9	15
	Rank	3	4	7	1	4	7	1	4

Table 4.14: IGD Ranking for 10-objective DTLZ

Algorithm	**Result**	**10-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
MGPSO_{STD}	Wins	3	4	4	5	4	4	4	28
	Losses	3	3	3	2	3	0	5	19
	Difference	0	1	1	3	1	4	-1	9
	Rank	5	4	4	3	4	1	6	4
MGPSO_{R}	Wins	3	4	4	0	4	4	5	24
	Losses	3	3	3	4	3	0	3	19
	Difference	0	1	1	-4	1	4	2	5
	Rank	5	4	4	6	4	1	4	5
MGPSO_{RI}	Wins	3	4	4	0	4	4	5	24
	Losses	3	3	3	4	3	0	3	19
	Difference	0	1	1	-4	1	4	2	5
	Rank	5	4	4	6	4	1	4	5
PMGPSO_{STD}	Wins	0	7	7	5	7	4	7	37
	Losses	7	0	0	2	0	0	0	9
	Difference	-7	7	7	3	7	4	7	28
	Rank	8	1	1	3	1	1	1	1

Table 4.14: IGD Ranking for 10-objective DTLZ (continue)

Algorithm	Result	10-objective DTLZ 1	2	3	4	5	6	7	Overall
$PMGPSO_R$	Wins	0	7	7	0	7	4	7	32
	Losses	7	0	0	4	0	0	0	11
	Difference	-7	7	7	-4	7	4	7	21
	Rank	8	1	1	6	1	1	1	2
$PMGPSO_{RI}$	Wins	0	7	7	0	7	4	7	32
	Losses	7	0	0	4	0	0	0	11
	Difference	-7	7	7	-4	7	4	7	21
	Rank	8	1	1	6	1	1	1	2
CDAS-SMPSO	Wins	3	1	0	0	1	0	0	5
	Losses	2	7	9	2	8	8	9	45
	Difference	1	-6	-9	-2	-7	-8	-9	-40
	Rank	4	8	10	5	9	10	10	10
KnEA	Wins	8	0	1	0	0	0	2	11
	Losses	0	7	8	4	9	7	6	41
	Difference	8	-7	-7	-4	-9	-7	-4	-30
	Rank	1	9	9	6	10	9	7	9
MOEA/DD	Wins	8	0	2	8	2	1	1	22
	Losses	0	8	7	0	6	6	8	35
	Difference	8	-8	-5	8	-4	-5	-7	-13
	Rank	1	10	8	1	7	8	9	8
NSGA-III	Wins	6	3	3	8	2	2	2	26
	Losses	2	6	6	0	6	6	6	32
	Difference	4	-3	-3	8	-4	-4	-4	-6
	Rank	3	7	7	1	7	7	7	7

Table 4.15: IGD Ranking for 15-objective DTLZ

Algorithm	**Result**	**15-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$MGPSO_{STD}$	Wins	3	4	3	5	3	4	2	24
	Losses	4	0	4	0	3	0	5	16
	Difference	-1	4	-1	5	0	4	-3	8
	Rank	5	1	5	1	5	1	6	4
$MGPSO_{R}$	Wins	3	4	3	0	3	4	5	22
	Losses	4	0	4	2	3	0	3	16
	Difference	-1	4	-1	-2	0	4	2	6
	Rank	5	1	5	7	5	1	4	6
$MGPSO_{RI}$	Wins	3	4	3	0	4	4	5	23
	Losses	4	0	4	2	3	0	3	16
	Difference	-1	4	-1	-2	1	4	2	7
	Rank	5	1	5	7	4	1	4	5
$PMGPSO_{STD}$	Wins	0	4	6	5	7	4	7	33
	Losses	7	0	1	0	0	0	0	8
	Difference	-7	4	5	5	7	4	7	25
	Rank	8	1	2	1	1	1	1	1
$PMGPSO_{R}$	Wins	0	4	6	1	7	4	7	29
	Losses	7	0	1	2	0	0	0	10
	Difference	-7	4	5	-1	7	4	7	19
	Rank	8	1	2	6	1	1	1	2
$PMGPSO_{RI}$	Wins	0	4	6	0	7	4	7	28
	Losses	7	0	1	2	0	0	0	10
	Difference	-7	4	5	-2	7	4	7	18
	Rank	8	1	2	7	1	1	1	3

Table 4.15: IGD Ranking for 15-objective DTLZ (continue)

Algorithm	Result	15-objective DTLZ							Overall
		1	2	3	4	5	6	7	
CDAS-SMPSO	Wins	7	2	0	0	1	0	0	10
	Losses	1	6	9	0	8	7	9	40
	Difference	6	-4	-9	0	-7	-7	-9	-30
	Rank	2	7	10	3	9	9	10	9
KnEA	Wins	7	0	1	0	0	0	2	10
	Losses	1	8	8	3	9	7	5	41
	Difference	6	-8	-7	-3	-9	-7	-3	-31
	Rank	2	9	9	10	10	9	6	10
MOEA/DD	Wins	9	0	2	0	2	0	1	14
	Losses	0	8	7	0	7	6	8	36
	Difference	9	-8	-5	0	-5	-6	-7	-22
	Rank	1	9	8	3	8	8	9	8
NSGA-III	Wins	6	2	9	0	3	2	2	24
	Losses	3	6	0	0	4	6	5	24
	Difference	3	-4	9	0	-1	-4	-3	0
	Rank	4	7	1	3	7	7	6	7

Table 4.16: IGD Ranking for 3-objective WFG

Algorithm	Result	3-objective WFG									Overall
		1	2	3	4	5	6	7	8	9	
$MGPSO_{STD}$	Wins	2	0	5	0	2	2	1	0	1	13
	Losses	4	7	4	7	3	7	7	7	7	53
	Difference	-2	-7	1	-7	-1	-5	-6	-7	-6	-40
	Rank	5	8	5	8	4	8	8	8	8	8

Table 4.16: IGD Ranking for 3-objective WFG (continue)

Algorithm	**Result**	**3-objective WFG** 1	2	3	4	5	6	7	8	9	**Overall**
MGPSO_R	Wins	0	0	3	0	1	0	0	0	0	4
	Losses	8	7	5	7	3	8	8	7	7	60
	Difference	-8	-7	-2	-7	-2	-8	-8	-7	-7	-56
	Rank	9	8	6	8	7	9	10	8	9	9
MGPSO_{RI}	Wins	0	0	3	0	1	0	0	0	0	4
	Losses	8	7	5	7	3	8	7	7	8	60
	Difference	-8	-7	-2	-7	-2	-8	-7	-7	-8	-56
	Rank	9	8	6	8	7	9	9	8	10	9
PMGPSO_{STD}	Wins	2	5	7	3	9	5	3	7	8	49
	Losses	4	0	0	1	0	0	6	0	0	11
	Difference	-2	5	7	2	9	5	-3	7	8	38
	Rank	5	1	1	4	1	1	7	1	1	2
PMGPSO_R	Wins	2	5	7	3	7	5	5	7	7	48
	Losses	4	0	0	1	1	0	2	0	2	10
	Difference	-2	5	7	2	6	5	3	7	5	38
	Rank	5	1	1	4	2	1	3	1	3	2
PMGPSO_{RI}	Wins	2	5	7	4	7	5	4	7	8	49
	Losses	4	0	0	1	1	0	2	0	0	8
	Difference	-2	5	7	3	6	5	2	7	8	41
	Rank	5	1	1	2	2	1	4	1	1	1
CDAS-SMPSO	Wins	7	3	1	4	1	4	4	4	3	31
	Losses	0	5	7	1	6	5	3	3	6	36
	Difference	7	-2	-6	3	-5	-1	1	1	-3	-5
	Rank	1	7	8	2	9	6	6	4	7	6

Table 4.16: IGD Ranking for 3-objective WFG (continue)

Algorithm	Result	3-objective WFG 1	2	3	4	5	6	7	8	9	Overall
KnEA	Wins	6	3	0	3	0	3	4	3	4	26
	Losses	3	4	9	3	9	6	2	6	5	47
	Difference	3	-1	-9	0	-9	-3	2	-3	-1	-21
	Rank	4	6	10	7	10	7	4	7	6	7
MOEA/DD	Wins	7	4	1	3	2	5	8	4	6	40
	Losses	0	0	7	1	3	0	0	3	3	17
	Difference	7	4	-6	2	-1	5	8	1	3	23
	Rank	1	5	8	4	4	1	1	4	4	5
NSGA-III	Wins	7	5	6	9	2	5	8	4	5	51
	Losses	0	0	3	0	3	0	0	3	4	13
	Difference	7	5	3	9	-1	5	8	1	1	38
	Rank	1	1	4	1	4	1	1	4	5	2

Table 4.17: IGD Ranking for 5-objective WFG

Algorithm	Result	5-objective WFG 1	2	3	4	5	6	7	8	9	Overall
$MGPSO_{STD}$	Wins	0	0	4	0	4	2	0	0	0	10
	Losses	4	5	4	7	3	7	7	7	7	51
	Difference	-4	-5	0	-7	1	-5	-7	-7	-7	-41
	Rank	5	6	5	8	4	8	8	8	8	8
$MGPSO_R$	Wins	0	0	2	0	3	0	0	0	0	5
	Losses	4	5	6	7	3	8	7	7	7	54
	Difference	-4	-5	-4	-7	0	-8	-7	-7	-7	-49
	Rank	5	6	7	8	6	9	8	8	8	10

Table 4.17: IGD Ranking for 5-objective WFG (continue)

Algorithm	**Result**	**5-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
MGPSO_{RI}	Wins	0	0	2	0	4	0	0	0	0	6
	Losses	4	5	6	7	3	8	7	7	7	54
	Difference	-4	-5	-4	-7	1	-8	-7	-7	-7	-48
	Rank	5	6	7	8	4	9	8	8	8	9
PMGPSO_{STD}	Wins	0	6	7	3	7	4	3	3	7	40
	Losses	4	0	0	4	0	4	4	4	0	20
	Difference	-4	6	7	-1	7	0	-1	-1	7	20
	Rank	5	1	1	5	1	5	5	5	1	3
PMGPSO_{R}	Wins	0	6	7	3	7	3	3	3	7	39
	Losses	4	0	0	4	0	4	4	4	0	20
	Difference	-4	6	7	-1	7	-1	-1	-1	7	19
	Rank	5	1	1	5	1	6	5	5	1	4
PMGPSO_{RI}	Wins	0	6	7	3	7	3	3	3	7	39
	Losses	4	0	0	4	0	5	4	4	0	21
	Difference	-4	6	7	-1	7	-2	-1	-1	7	18
	Rank	5	1	1	5	1	7	5	5	1	5
CDAS-SMPSO	Wins	8	6	4	8	3	8	8	6	4	55
	Losses	0	0	4	0	5	1	1	2	5	18
	Difference	8	6	0	8	-2	7	7	4	-1	37
	Rank	1	1	5	1	7	2	2	3	6	2
KnEA	Wins	6	0	0	6	1	6	6	6	3	34
	Losses	3	5	9	3	8	2	2	2	6	40
	Difference	3	-5	-9	3	-7	4	4	4	-3	-6
	Rank	4	6	10	4	9	3	3	3	7	7

Table 4.17: IGD Ranking for 5-objective WFG (continue)

Algorithm	Result	1	2	3	4	5	6	7	8	9	Overall
MOEA/DD	Wins	7	0	1	7	0	6	6	8	5	40
	Losses	1	5	8	2	9	2	2	1	4	34
	Difference	6	-5	-7	5	-9	4	4	7	1	6
	Rank	3	6	9	3	10	3	3	2	5	6
NSGA-III	Wins	7	5	6	8	2	9	9	9	6	61
	Losses	0	4	3	0	7	0	0	0	3	17
	Difference	7	1	3	8	-5	9	9	9	3	44
	Rank	2	5	4	1	8	1	1	1	4	1

Table 4.18: IGD Ranking for 8-objective WFG

Algorithm	Result	1	2	3	4	5	6	7	8	9	Overall
MGPSO_{STD}	Wins	0	1	3	0	3	1	1	1	1	11
	Losses	3	5	4	7	3	6	6	6	5	45
	Difference	-3	-4	-1	-7	0	-5	-5	-5	-4	-34
	Rank	4	7	5	8	4	7	7	7	6	7
MGPSO_{R}	Wins	0	1	3	0	3	1	1	1	1	11
	Losses	3	5	4	7	3	6	6	6	5	45
	Difference	-3	-4	-1	-7	0	-5	-5	-5	-4	-34
	Rank	4	7	5	8	4	7	7	7	6	7
MGPSO_{RI}	Wins	0	1	3	0	3	1	1	1	1	11
	Losses	3	5	4	7	3	6	6	6	5	45
	Difference	-3	-4	-1	-7	0	-5	-5	-5	-4	-34
	Rank	4	7	5	8	4	7	7	7	6	7

Table 4.18: IGD Ranking for 8-objective WFG (continue)

Algorithm	Result	8-objective WFG 1	2	3	4	5	6	7	8	9	Overall
$PMGPSO_{STD}$	Wins	0	7	6	3	7	4	4	4	7	42
	Losses	3	0	2	4	0	3	3	3	0	18
	Difference	-3	7	4	-1	7	1	1	1	7	24
	Rank	4	1	4	5	1	4	4	4	1	4
$PMGPSO_{R}$	Wins	0	6	7	3	7	4	4	4	7	42
	Losses	3	0	0	4	0	3	3	3	0	16
	Difference	-3	6	7	-1	7	1	1	1	7	26
	Rank	4	3	1	5	1	4	4	4	1	3
$PMGPSO_{RI}$	Wins	0	7	7	3	7	4	4	4	7	43
	Losses	3	0	0	4	0	3	3	3	0	16
	Difference	-3	7	7	-1	7	1	1	1	7	27
	Rank	4	1	1	5	1	4	4	4	1	2
CDAS-SMPSO	Wins	8	0	6	9	1	7	7	7	1	46
	Losses	0	9	0	0	7	1	2	0	5	24
	Difference	8	-9	6	9	-6	6	5	7	-4	22
	Rank	1	10	3	1	8	2	3	1	6	5
KnEA	Wins	7	4	0	6	0	0	0	0	0	17
	Losses	1	4	8	3	9	9	9	9	9	61
	Difference	6	0	-8	3	-9	-9	-9	-9	-9	-44
	Rank	3	5	10	4	10	10	10	10	10	10
MOEA/DD	Wins	0	1	0	7	1	7	8	7	5	36
	Losses	3	4	7	1	7	1	0	0	3	26
	Difference	-3	-3	-7	6	-6	6	8	7	2	10
	Rank	4	6	9	2	8	2	1	1	4	6

Table 4.18: IGD Ranking for 8-objective WFG (continue)

Algorithm	**Result**	**8-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
NSGA-III	Wins	7	6	1	7	3	9	8	7	5	53
	Losses	0	2	7	1	3	0	0	0	3	16
	Difference	7	4	-6	6	0	9	8	7	2	37
	Rank	2	4	8	2	4	1	1	1	4	1

Table 4.19: IGD Ranking for 10-objective WFG

Algorithm	**Result**	**10-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
$MGPSO_{STD}$	Wins	0	2	3	0	4	2	0	0	4	15
	Losses	3	5	3	5	3	5	6	6	3	39
	Difference	-3	-3	0	-5	1	-3	-6	-6	1	-24
	Rank	4	6	5	9	4	6	8	10	4	9
$MGPSO_{R}$	Wins	0	2	3	0	4	2	0	0	4	15
	Losses	3	5	3	5	3	5	6	5	3	38
	Difference	-3	-3	0	-5	1	-3	-6	-5	1	-23
	Rank	4	6	5	9	4	6	8	9	4	7
$MGPSO_{RI}$	Wins	0	2	3	0	4	2	0	1	4	16
	Losses	3	5	3	4	3	5	6	5	3	37
	Difference	-3	-3	0	-4	1	-3	-6	-4	1	-21
	Rank	4	6	5	6	4	6	8	8	4	6
$PMGPSO_{STD}$	Wins	0	6	6	2	7	5	3	3	7	39
	Losses	3	0	0	4	0	2	3	3	0	15
	Difference	-3	6	6	-2	7	3	0	0	7	24
	Rank	4	3	1	5	1	3	4	5	1	3

Table 4.19: IGD Ranking for 10-objective WFG (continue)

Algorithm	**Result**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Overall**
		10-objective WFG									
$PMGPSO_R$	Wins	0	7	6	0	7	5	3	4	7	39
	Losses	3	0	0	4	0	2	3	2	0	14
	Difference	-3	7	6	-4	7	3	0	2	7	25
	Rank	4	1	1	6	1	3	4	3	1	2
$PMGPSO_{RI}$	Wins	0	7	6	0	7	5	3	3	7	38
	Losses	3	0	0	4	0	2	3	2	0	14
	Difference	-3	7	6	-4	7	3	0	1	7	24
	Rank	4	1	1	6	1	3	4	4	1	3
CDAS-SMPSO	Wins	8	0	3	9	2	8	9	8	0	47
	Losses	0	8	0	0	6	0	0	0	7	21
	Difference	8	-8	3	9	-4	8	9	8	-7	26
	Rank	1	9	4	1	7	1	1	1	9	1
KnEA	Wins	7	5	0	6	1	0	0	0	0	19
	Losses	1	4	7	1	8	8	3	2	8	42
	Difference	6	1	-7	5	-7	-8	-3	-2	-8	-23
	Rank	3	5	9	2	9	9	7	6	10	7
MOEA/DD	Wins	0	0	0	6	0	0	7	0	2	15
	Losses	3	8	8	1	9	8	2	2	6	47
	Difference	-3	-8	-8	5	-9	-8	5	-2	-4	-32
	Rank	4	9	10	2	10	9	3	6	7	10
NSGA-III	Wins	7	6	1	6	2	8	8	8	1	47
	Losses	0	2	7	1	6	0	1	0	6	23
	Difference	7	4	-6	5	-4	8	7	8	-5	24
	Rank	2	4	8	2	7	1	2	1	8	3

Table 4.20: IGD Ranking for 15-objective WFG

Algorithm	**Result**	**15-objective WFG 1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Overall**
$MGPSO_{STD}$	Wins	0	1	3	0	4	0	0	0	4	12
	Losses	1	5	1	7	3	5	6	7	3	38
	Difference	-1	-4	2	-7	1	-5	-6	-7	1	-26
	Rank	5	7	5	8	4	10	9	10	4	10
$MGPSO_{R}$	Wins	0	1	3	0	4	0	0	0	4	12
	Losses	1	5	1	7	3	3	5	6	3	34
	Difference	-1	-4	2	-7	1	-3	-5	-6	1	-22
	Rank	5	7	5	8	4	8	8	8	4	8
$MGPSO_{RI}$	Wins	0	1	3	0	4	0	0	0	4	12
	Losses	1	5	1	7	3	3	6	6	3	35
	Difference	-1	-4	2	-7	1	-3	-6	-6	1	-23
	Rank	5	7	5	8	4	8	9	8	4	9
$PMGPSO_{STD}$	Wins	0	6	3	3	7	1	0	1	7	28
	Losses	1	0	0	4	0	2	4	4	0	15
	Difference	-1	6	3	-1	7	-1	-4	-3	7	13
	Rank	5	1	2	5	1	4	7	7	1	4
$PMGPSO_{R}$	Wins	0	6	6	3	7	1	2	3	7	35
	Losses	1	0	0	4	0	2	4	4	0	15
	Difference	-1	6	6	-1	7	-1	-2	-1	7	20
	Rank	5	1	1	5	1	4	6	5	1	2
$PMGPSO_{RI}$	Wins	0	6	3	3	7	3	3	3	7	35
	Losses	1	0	0	4	0	2	4	4	0	15
	Difference	-1	6	3	-1	7	1	-1	-1	7	20
	Rank	5	1	2	5	1	3	5	5	1	2

Table 4.20: IGD Ranking for 15-objective WFG (continue)

Algorithm	Result	15-objective WFG 1	2	3	4	5	6	7	8	9	Overall
CDAS-SMPSO	Wins	0	0	3	9	2	9	9	9	2	43
	Losses	0	9	0	0	6	0	0	0	6	21
	Difference	0	-9	3	9	-4	9	9	9	-4	22
	Rank	2	10	2	1	7	1	1	1	7	1
KnEA	Wins	6	1	0	6	1	0	7	6	2	29
	Losses	0	4	7	1	8	2	1	1	6	30
	Difference	6	-3	-7	5	-7	-2	6	5	-4	-1
	Rank	1	6	8	3	9	6	2	3	7	6
MOEA/DD	Wins	0	4	0	6	0	0	6	6	0	22
	Losses	0	4	7	2	9	2	2	2	8	36
	Difference	0	0	-7	4	-9	-2	4	4	-8	-14
	Rank	2	5	8	4	10	6	4	4	9	7
NSGA-III	Wins	0	6	0	7	2	8	6	7	0	36
	Losses	0	0	7	1	6	1	1	1	8	25
	Difference	0	6	-7	6	-4	7	5	6	-8	11
	Rank	2	1	8	2	7	2	3	2	9	5

4.4.1 Hypervolume Discussion

The overall HV rankings for each algorithm, shown in tables 4.1 to 4.10, are analyzed and discussed next.

The MGPSO$_{STD}$ ranked first overall four times with regards to HV – more than any other algorithm investigated in this chapter. The MGPSO$_{STD}$ ranked first overall only with respect to HV, for the 3-, 8-, 10-, and the 15-objective DTLZ benchmark problems (tables 4.1 and 4.3 to 4.5). The MGPSO$_{STD}$ also ranked top-three overall eight times with respect to HV, for the DTLZ and WFG problems (tables 4.1 to 4.5, and 4.8 to 4.10). That is, the most top-three overall ranks with respect to HV obtained by any algorithm investigated in this chapter. The worst overall rank obtained by the MGPSO$_{STD}$ with respect to HV was seven, for the 5-objective WFG problems (table 4.7). Therefore, the

HV rankings indicate that the MGPSO$_{STD}$ is very competitive, even compared to algorithms developed specifically to solve MaOPs.

The MGPSO$_R$ was the only algorithm that did not rank first overall even once with respect to both HV and IGD. However, the MGPSO$_R$ ranked top-three overall seven times with respect to HV, for the DTLZ and WFG problems (tables 4.1, 4.4, 4.5, and 4.7 to 4.10). That is, the second-most top-three overall ranked algorithm with respect to HV. The best overall rank obtained by the MGPSO$_R$ with respect to HV was two, for the 15-objective DTLZ and 8-, 10-, and 15-objective WFG problems (tables 4.5 and 4.8 to 4.10). The worst overall HV ranking obtained by the MGPSO$_R$ was five, for the 8-objective DTLZ problems (table 4.3). Therefore, the MGPSO$_R$ performed and scaled competitively with respect to HV.

The MGPSO$_{RI}$ ranked first overall twice in terms of HV; that is, for the 8- and 10-objective WFG problems (tables 4.8 and 4.9). The MGPSO$_{RI}$ ranked top-three overall six times with respect to HV for the DTLZ and WFG problems (tables 4.2, 4.4 to 4.6, 4.8, and 4.9). The worst overall HV ranking for the MGPSO$_{RI}$ was five, for the 5-objective WFG problems (table 4.7). Therefore, the MGPSO$_{RI}$ performed and scaled very competitively with respect to HV. The performance of the MGPSO$_R$ and the MGPSO$_{RI}$ was comparable with respect to HV.

The best overall HV rank received by the PMGPSO$_{STD}$ was four, for the 15-objective DTLZ problems (table 4.5) and the 10-objective WFG problems (table 4.9). The PMGPSO$_{STD}$ was the only algorithm that did not rank worst overall even once (for both performance measures). The worst overall HV ranking for the PMGPSO$_{STD}$ was seven, for the 3-objective DTLZ problems (table 4.1). The PMGPSO$_{STD}$ ranked toward the middle overall often with respect to HV. Therefore, relative to the other investigated algorithms, the PMGPSO$_{STD}$ did not perform competitively with regards to HV.

The best overall HV rank obtained by the PMGPSO$_R$ was five, for the 15-objective DTLZ problems and the 10-objective WFG problems (tables 4.5 and 4.9). The PMGPSO$_R$ ranked worst overall twice in terms of HV; that is, for the 3- and 5-objective WFG problems (tables 4.6 and 4.7). The PMGPSO$_R$ ranked toward the end overall often with respect to HV. Therefore, the PMGPSO$_R$ did not perform competitively with regards to HV.

The best overall HV ranking for the PMGPSO$_{RI}$ was five, for the 15-objective DTLZ and WFG problems (tables 4.5 and 4.10). The PMGPSO$_{RI}$ only received the worst overall rank once in terms of HV, for the 3-objective WFG problems (table 4.6). The PMGPSO$_R$ ranked toward the end overall often with respect to HV. Therefore, the PMGPSO$_{RI}$ did not perform competitively with respect to HV. The performance of the PMGPSO$_{RI}$ and the PMGPSO$_R$ was similar in terms of HV. The PMGPSO$_{RI}$ and the PMGPSO$_R$ performed marginally worse than the PMGPSO$_{STD}$ in terms of HV.

The best overall rank obtained by the CDAS-SMPSO algorithm in terms of HV was two, for the 3- and 5-objective WFG problems (tables 4.6 and 4.7).

That is, the only two top-three overall ranks obtained by the CDAS-SMPSO algorithm with respect to HV. The CDAS-SMPSO algorithm ranked worst overall with respect to HV once. That is, for the 3-objective DTLZ problems (table 4.1). Therefore, the CDAS-SMPSO algorithm performed inconsistently with regards to HV.

The KnEA performed best overall twice in terms of HV; i.e., for the 3- and 5-objective WFG problems (tables 4.6 and 4.7). The KnEA only ranked top-three overall three times, for the DTLZ problems with respect to HV (tables 4.1, 4.6 and 4.7). The KnEA ranked last overall more than any other algorithm. The KnEA ranked worst overall four times with respect to HV; that is, for the 5- and 8-objective DTLZ and 8- and 15-objective WFG problems (tables 4.2, 4.3, 4.8, and 4.10). The few cases for which the KnEA did perform best or top-three overall with respect to HV were for problems with few objectives. The KnEA also performed inconsistently in terms of HV. Therefore, the KnEA performed worst overall with regards to HV.

The MOEA/DD received the top overall rank only once, for the 5-objective DTLZ problems with respect to HV (table 4.2). The MOEA/DD only ranked among the top-three overall one other time; i.e., for the 8-objective DTLZ problems in terms of HV (table 4.3). The MOEA/DD had the worst overall rank associated with it twice with respect to HV; that is, for the 10-objective DTLZ and WFG problems (tables 4.4 and 4.9). Therefore, the MOEA/DD performed inconsistently with respect to HV.

The NSGA-III ranked first overall once in terms of HV, for the 15-objective WFG problems (table 4.10). The NSGA-III ranked in the top-three overall three times with respect to HV; that is, for the WFG problems (tables 4.7, 4.8, and 4.10. The NSGA-III received the worst overall rank with respect to HV for only the 15-objective DTLZ problems (table 4.5). Therefore, the NSGA-III performed somewhat competitively with regards to HV.

4.4.2 Inverted Generational Distance Discussion

The overall IGD rankings for each algorithm, shown in tables 4.11 to 4.20, are analyzed and discussed next.

The best overall rank obtained by the MGPSO$_{STD}$ in terms of IGD was four, for the 10- and 15-objective DTLZ problems (tables 4.14 and 4.15). The MGPSO$_{STD}$ ranked worst overall only once; that is, for the 15-objective WFG problems with respect to IGD (table 4.20). The MGPSO$_{STD}$ ranked towards the middle and bottom overall often with regards to IGD. Therefore, the MGPSO$_{STD}$ did not scale to many-objectives competitively with respect to IGD.

The MGPSO$_R$ was the only algorithm that did not rank first overall even once. The best overall rank obtained by the MGPSO$_R$ with respect to IGD was five, for the 10-objective DTLZ problems (table 4.14). The MGPSO$_R$ ranked worst overall two times in terms of IGD; that is, for the 3- and 5-objective

WFG problems (tables 4.16 and 4.17). The MGPSO$_R$ ranked towards the middle and bottom overall often with regards to IGD. Therefore, in terms of IGD, the MGPSO$_R$ did not perform competitively.

The best overall rank received by the MGPSO$_{RI}$ in terms of IGD was five, for the 10-objective DTLZ problems (table 4.14). The MGPSO$_{RI}$ received the worst overall ranking in terms of IGD for the 3-objective WFG problems only (table 4.16). The MGPSO$_{RI}$ ranked towards the middle and bottom overall often with regards to IGD. Therefore, the MGPSO$_{RI}$ did not scale competitively in terms of the IGD. The performance of the MGPSO$_{STD}$, the MGPSO$_R$, and the MGPSO$_{RI}$ was similar with respect to IGD.

The PMGPSO$_{STD}$ ranked first overall three times with respect to IGD; that is, the most compared to any other algorithm investigated in this chapter. The PMGPSO$_{STD}$ ranked first overall with respect to IGD, for the DTLZ problems with 8, 10, and 15 objectives (tables 4.13 to 4.15). The PMGPSO$_{STD}$ ranked top-three overall eight times with respect to IGD, for the DTLZ and WFG problems (tables 4.11 to 4.17, and 4.19). That is, the second-most top-three overall ranked algorithm with respect to IGD. Note again that the PMGPSO$_{STD}$ was the only algorithm that did not rank worst overall even once for both performance measures. The worst overall IGD ranking for the PMGPSO$_{STD}$ was only four, for the 8- and 15-objective WFG problems (tables 4.18 and 4.20). Therefore, the PMGPSO$_{STD}$ scaled to many-objectives best with regards to IGD.

The PMGPSO$_R$ ranked first overall only once; that is, for the 3-objective DTLZ problems with respect to IGD (table 4.11). However, the PMGPSO$_R$ ranked top-three overall nine times in terms of IGD, for the DTLZ and WFG problems; that is, the most top-three overall ranks obtained by any of the algorithms (tables 4.11 to 4.16, and 4.18 to 4.20). The worst overall IGD ranking for the PMGPSO$_R$ was only four, the 5-objective WFG problems (table 4.17). Therefore, the PMGPSO$_R$ scaled to many-objectives well with respect to IGD.

The PMGPSO$_{RI}$ ranked first overall twice with respect to IGD, for the 5-objective DTLZ problems (table 4.12) and the 3-objective WFG problems (table 4.16). The PMGPSO$_{RI}$ also ranked top-three overall nine times with respect to IGD, for the WFG and DTLZ problems (tables 4.11 to 4.16 and 4.18 to 4.20). That is, the most top-three overall ranks received by any of the algorithms (together with the PMGPSO$_R$). The worst overall IGD ranking for the PMGPSO$_{RI}$ was five, for the 5-objective WFG problems (table 4.17). Therefore, the PMGPSO$_{RI}$ scaled to many-objectives very competitively with respect to IGD. The performance of the PMGPSO$_{STD}$, the PMGPSO$_R$, and the PMGPSO$_{RI}$ was comparable in terms of IGD.

The CDAS-SMPSO algorithm ranked first overall twice with regards to IGD; that is, for the 10- and 15-objective WFG problems (tables 4.19 and 4.20). The CDAS-SMPSO algorithm ranked top-three overall three times with respect to IGD, for the WFG problems (tables 4.17, 4.19, and 4.20). The CDAS-

SMPSO algorithm ranked worst overall with respect to IGD three times. That is, for the 3-, 5-, and 10-objective DTLZ problems (tables 4.11, 4.12, and 4.14). Therefore, even with a few promising ranks, the CDAS-SMPSO algorithm performed very inconsistently in terms of IGD.

The best overall rank received by the KnEA with respect to IGD was six, for the 3-objective DTLZ problems and the 15-objective WFG problems (tables 4.11 and 4.20). The KnEA ranked worst overall more than any other algorithm. The KnEA ranked worst overall four times with respect to IGD; that is, for the 5-, 8-, and 15-objective DTLZ and the 8-objective WFG problems (tables 4.12, 4.13, 4.15, and 4.18). Therefore, the KnEA scaled the worst overall in terms of IGD.

The best overall IGD rank received by the MOEA/DD was four, obtained for the 3-objective DTLZ problems (table 4.11). The MOEA/DD had the worst overall rank associated with it once with respect to IGD; that is, for the 10-objective WFG problems (table 4.19). The MOEA/DD ranked toward the middle overall often with respect to IGD. Therefore, relative to the other investigated algorithms, the MOEA/DD did not perform competitively with respect to IGD.

The NSGA-III ranked first overall twice in terms of IGD, for the 5- and 8-objective WFG problems (tables 4.17 and 4.18). The NSGA-III ranked in the top-three overall four times with respect to IGD; that is, for the WFG problems (tables 4.16 to 4.19). The worst overall rank obtained by the NSGA-III with respect to IGD was seven, for the 10- and 15-objective DTLZ problems (tables 4.14 and 4.15). Therefore, the NSGA-III performed competitively in terms of IGD.

4.4.3 General Discussion

Some general findings with regards to tables 4.1 to 4.20 are discussed next.

The partial-dominance approach (the PMGPSO algorithm) improved the scalability of the MGPSO algorithm in terms of IGD, but not in terms of HV. Remember that the PMGPSO algorithm randomly selects three objectives each time the partial-dominance relation is applied; that is, whenever an insert into the archive is attempted. Sato *et al.* [133] showed that the interval at which new objectives are chosen for the partial-dominance relation has a significant impact on the resulting HV. Note, however, that benchmark MaOPs of a different kind was solved in [133]. Helbig and Engelbrecht [52, 69] also investigated larger intervals with some success for GAs and to a lesser extent for PSO algorithms. Also, note that the HV performance measure was not included in [52, 69]. Therefore, future research is required to investigate whether randomly selecting three objectives at different iteration intervals will lead to better PMGPSO algorithm performance (especially with regards to HV).

The MGPSO$_{STD}$ scaled to many-objectives surprisingly well in terms of HV, despite using the struggling Pareto-dominance relation. Recall from Chapter 3 that each particle in the MGPSO algorithm utilizes a personal best guide, a neighbourhood best guide, and an archive guide to help guide the swarm towards the POF. The personal best and neighbourhood best positions are updated using the corresponding objective function fitness value as a result of the multi-swarm approach employed by the MGPSO algorithm. The archive guide is selected as the least crowded solution from a random tournament of size three. Therefore, the multi-swarm approach may be an implicit mechanism to help overcome the degrading Pareto-dominance relation since each subswarm focuses on optimizing a specific objective which in turn promotes exploitation for each objective individually. It is also possible that the MGPSO algorithm scaled decently because the Pareto-dominance relation is only used to update the archive and does not have that strong of an effect on the selection of the search guides.

The use of different dynamic archive balance coefficient update strategies did not improve the performance (i.e. scalability) of the MGPSO algorithm nor the PMGPSO algorithm. The original STD approach was deemed the superior option for both the MGPSO algorithm and the PMGPSO algorithm. Future research will investigate combining the MGPSO$_{STD}$ and the PMGPSO$_{STD}$ in some way because the former performed best with respect to HV and the latter with respect to IGD. Note, however, that the MGPSO algorithm and PMGPSO algorithm variants, using the R and the RI, were still very competitive.

The results, discussed in the sections above, are summarized in tables 4.21 and 4.22. The tables contain the number of overall best, overall top-three, and overall worst ranks obtained by each algorithm with respect to either HV or IGD. Tables 4.23 and 4.24 contain the average overall rank obtained by each algorithm for each benchmark problem suite for each number of objectives with respect to either HV or IGD.

4.5 Summary

This chapter discussed the partial-dominance approach and introduced the partial-dominance multi-guide particle swarm optimization (PMGPSO) algorithm. The PMGPSO algorithm uses the partial-dominance relation, which modifies the Pareto-dominance relation in order to increase the selection pressure towards the POF during MaOO. The scalability of the multi-guide particle swarm optimization (MGPSO) algorithm [137], the PMGPSO algorithm, and a number of benchmark algorithms were empirically investigated on a set of benchmark problems by calculating, and statistically analyzing, the inverted generational distance (IGD) [25, 128] and hypervolume (HV) [181] performance measure values that were calculated on the normalized solutions without outliers. The benchmark algorithms included the controlling

Table 4.21: HV Ranking Summary

Algorithm	**Number of overall best HV ranks**	**Number of overall HV ranks ≤ 3**	**Number of overall worst HV ranks**
MGPSO$_{STD}$	4	8	0
MGPSO$_{R}$	0	7	0
MGPSO$_{RI}$	2	6	0
PMGPSO$_{STD}$	0	0	0
PMGPSO$_{R}$	0	0	2
PMGPSO$_{RI}$	0	0	1
CDAS-SMPSO	0	2	1
KnEA	2	3	4
MOEA/DD	1	2	2
NSGA-III	1	3	1

Table 4.22: IGD Ranking Summary

Algorithm	**Number of overall best IGD ranks**	**Number of overall IGD ranks ≤ 3**	**Number of overall worst IGD ranks**
MGPSO$_{STD}$	0	0	1
MGPSO$_{R}$	0	0	2
MGPSO$_{RI}$	0	0	1
PMGPSO$_{STD}$	3	8	0
PMGPSO$_{R}$	1	9	0
PMGPSO$_{RI}$	2	9	0
CDAS-SMPSO	2	3	3
KnEA	0	0	4
MOEA/DD	0	0	1
NSGA-III	2	4	0

dominance area of solutions speed constraint multi-objective particle swarm optimization (CDAS-SMPSO) [33] algorithm, the knee-point driven evolutionary algorithm (KnEA) [174], the many-objective evolutionary algorithm based on dominance and decomposition (MOEA/DD) [95], and the reference-point based many-objective non-dominated sorting genetic algorithm (NSGA-III) [38]. The benchmark functions were configured with 3, 5, 8, 10, and 15 objectives to test algorithm scalability. The number of decision variables was fixed at 30. Three different archive balance coefficient update strategies were also investigated with the aim of improving scalability. These included the standard static archive balance coefficient update strategy (STD), the random dynamic archive balance coefficient update strategy (R), and the random per

Table 4.23: HV Ranking Averages

Algorithm	Benchmark Problems	n_m 3	5	8	10	15	Average
$MGPSO_{STD}$	DTLZ	1	3	1	1	1	1.4
	WFG	5	7	3	2	3	4
	Average Rank	3	5	2	1.5	2	2.7
$MGPSO_{R}$	DTLZ	3	4	5	3	2	3.4
	WFG	4	3	2	2	2	2.6
	Average Rank	3.5	3.5	3.5	2.5	2	3
$MGPSO_{RI}$	DTLZ	4	2	4	2	4	3.2
	WFG	3	5	1	1	4	2.8
	Average Rank	3.5	3.5	2.5	1.5	4	3
$PMGPSO_{STD}$	DTLZ	7	6	6	5	4	5.6
	WFG	8	8	5	4	5	6
	Average Rank	7.5	7	5.5	4.5	4.5	5.8
$PMGPSO_{R}$	DTLZ	8	8	8	8	5	7.4
	WFG	9	10	8	5	7	7.8
	Average Rank	8.5	9	8	6.5	6	7.6
$PMGPSO_{RI}$	DTLZ	9	7	9	9	5	7.8
	WFG	9	9	7	6	5	7.2
	Average Rank	9	8	8	7.5	5	7.5
CDAS-SMPSO	DTLZ	10	9	7	6	9	8.2
	WFG	2	2	4	7	9	4.8
	Average Rank	6	5.5	5.5	6.5	9	6.5
KnEA	DTLZ	2	10	10	7	8	7.4
	WFG	1	1	10	9	10	6.2
	Average Rank	1.5	5.5	10	8	9	6.8
MOEA/DD	DTLZ	5	1	2	10	7	5
	WFG	7	6	9	10	8	8
	Average Rank	6	3.5	5.5	10	7.5	6.5
NSGA-III	DTLZ	6	5	3	4	10	5.6
	WFG	6	3	6	8	1	4.8
	Average Rank	6	4	4.5	6	5.5	5.2

Table 4.24: IGD Ranking Averages

Algorithm	**Benchmark Problems**	n_m **3**	**5**	**8**	**10**	**15**	**Average**
MGPSO_{STD}	DTLZ	9	8	6	4	4	6.2
	WFG	8	8	7	9	10	8.4
	Average Rank	8.5	8	6.5	6.5	7	7.3
MGPSO_{R}	DTLZ	8	6	7	5	6	6.4
	WFG	9	10	7	7	8	8.2
	Average Rank	8.5	8	7	6	7	7.3
MGPSO_{RI}	DTLZ	7	7	7	5	5	6.2
	WFG	9	9	7	6	9	8
	Average Rank	8	8	7	5.5	7	7.1
PMGPSO_{STD}	DTLZ	3	1	1	1	1	1.4
	WFG	2	3	4	3	4	3.2
	Average Rank	2.5	2	2.5	2	2.5	2.3
PMGPSO_{R}	DTLZ	1	3	2	2	2	2
	WFG	2	4	3	2	2	2.6
	Average Rank	1.5	3.5	2.5	2	2	2.3
PMGPSO_{RI}	DTLZ	2	1	3	2	3	2.2
	WFG	1	5	2	3	2	2.6
	Average Rank	1.5	3	2.5	2.5	2.5	2.4
CDAS-SMPSO	DTLZ	10	10	9	10	9	9.6
	WFG	6	2	5	1	1	3
	Average Rank	8	6	7	5.5	5	6.3
KnEA	DTLZ	6	10	10	9	10	9
	WFG	7	7	10	7	6	7.4
	Average Rank	6.5	8.5	10	8	8	8.2
MOEA/DD	DTLZ	4	5	5	8	8	6
	WFG	5	6	6	10	7	6.8
	Average Rank	4.5	5.5	5.5	9	7.5	6.4
NSGA-III	DTLZ	5	4	4	7	7	5.4
	WFG	2	1	1	3	5	2.4
	Average Rank	3.5	2.5	2.5	5	6	3.9

particle dynamic archive balance coefficient update strategy (RI) [53]. The results were presented in tables and discussed.

The MGPSO algorithm using the STD (MGPSO$_{STD}$) scaled better in terms of HV than any other algorithm investigated in this chapter. The MGPSO algorithm using the R (MGPSO$_{R}$) and the MGPSO algorithm using the RI (MGPSO$_{RI}$) performed competitively in terms of HV, ranking top-three overall often. The PMGPSO algorithm using the STD (PMGPSO$_{STD}$) scaled best overall with regards to IGD. The PMGPSO$_{STD}$ also never performed worst. The PMGPSO algorithm using the R (PMGPSO$_{R}$) and the PMGPSO algorithm using the RI (PMGPSO$_{RI}$) performed competitively in terms of IGD, ranking top-three overall most often. The dynamic archive balance coefficient update strategies did not improve the scalability of MGPSO algorithm or the PMGPSO algorithm.

The partial-dominance approach used by the PMGPSO algorithm did improve the scalability of the MGPSO algorithm but only with regards to IGD when considering the number of top-three overall ranks. The partial-dominance approach slightly deteriorated algorithm scalability in terms of HV. This may be attributed to the interval size at which new objectives were randomly selected to be considered for the partial-dominance relation [52, 69, 133]. That is, the PMGPSO algorithm randomly reselects three objectives every time an insert into the archive is attempted. Randomly reselecting objectives at each iteration or after a certain number of iterations may improve algorithm performance [52, 69], especially with respect to HV [133].

In terms of the benchmark algorithms, the CDAS-SMPSO algorithm, the KnEA, and the MOEA/DD ranked worst overall more often compared to the other algorithms. The KnEA, however, scaled to many-objective optimization problems (MaOPs) the worst. The NSGA-III was the only algorithm that obtained top overall ranks both in terms of HV and IGD. The NSGA-III was very competitive compared to the other benchmark algorithms.

Chapter 5

Knee-point driven Multi-guide Particle Swarm Optimization

"You are the most influential person you will talk to all day."

— Zig Ziglar

This chapter proposes another new variation of the MGPSO algorithm, i.e., the knee-point driven multi-guide particle swarm optimization (KnMGPSO) algorithm. As sub-objectives, this chapter aims to evaluate the performance of the KnMGPSO algorithm in comparison with other algorithms; and to evaluate the effect of different archive balance coefficient update strategies for the KnMGPSO algorithm. This chapter also empirically compares the scalability of the KnMGPSO algorithm with that of the MGPSO algorithm and several other benchmark MaOO algorithms. More specifically, section 5.1 proposes knee-points as another approach to attempt to improve the scalability of the MGPSO algorithm. The KnMGPSO algorithm is presented in section 5.2, the empirical process followed is presented in section 5.3, and section 5.4 presents the results and discusses the findings. Finally, section 5.5 gives a summary of this chapter.

5.1 Knee-points Approach

A number of approaches have been proposed to enable algorithms to find regions or points of interest in the POF [109, 130, 152]. Oftentimes points referred to as knee-points are considered to be of interest. Knee-points within the current found front represent the solutions that are converging best within their immediate neighbourhood and are therefore useful for increasing the selection pressure to converge towards the POF [105]. Intuitively, a knee-point is a Pareto-optimal solution with maximum marginal rates of return, which means that a small improvement in one objective of such a solution is accompanied by a severe deterioration in at least one other objective. That is, minimally improving some objective while extremely degrading others cannot be justified in the absence of a decision-maker where all of the objectives are considered to be equally important. It can be said that knee-points represent the naturally preferable solutions within the POF when no domain knowledge is available.

Zhang *et. al.* [174] noted that the incorporation of knee-points during optimization results in a bias towards a higher HV. Maltese *et. al.* [105] stated that since the hypervolume metric is maximized if, and only if, the solution set consists entirely of all Pareto-optimal points [57], prioritizing hypervolume via knee-points should theoretically aid convergence towards the true Pareto front. Both Zhang *et. al.* [174] and Maltese *et. al.* [105] have successfully used knee-points for MaOO. For this reason, this study proposes knee-points as the second main viable solution to help the MGPSO algorithm scale to MaOPs. This work uses the same adaptive knee-points identification approach as [174] and [105]. Note that this approach was thoroughly discussed in section 3.1.2.1.

It is noted in [174] that, in some cases, the weighted distance between solutions can be superior to crowding distance for diversity purposes. Figure 5.1, taken from [174], illustrates a situation where if the crowding distance is used, neither solution B nor solution C will have the chance to win against other solutions. However, from the diversity (i.e. exploration) point of view, it would be helpful if either B or C can have a chance to win in the tournament for reproduction. By using the weighted distance operator, such solutions stand a chance to be considered. Note, however, that this thesis focusses on improving the lack of selection pressure experienced when primarily using the Pareto-dominance relation and not on improving diversity. In other words, the crowding distance operator has been sufficient so far in terms of solution diversity management and preservation for the MGPSO algorithm [136]. Future research will investigate the impact that weighted distance has on the performance of the MGPSO algorithm. Note, however, that the weighted distance approach has the drawback of introducing a parameter for the k-nearest neighbours [56] component, that ideally needs to be tuned since it determines how many of the closest solutions, in objective space, should be considered for the weighted distance calculation. In [174], three-nearest neighbours were used without stating whether other values were tested. In a more recent paper [105], no detail is given about the parameter setting for the number of neighbours considered for the weighted distance calculation.

5.2 Knee-point driven Multi-guide Particle Swarm Optimization

The Pareto-dominance relation breaks down as the number of objectives increases due to a lack of selection pressure in larger-dimensional objective

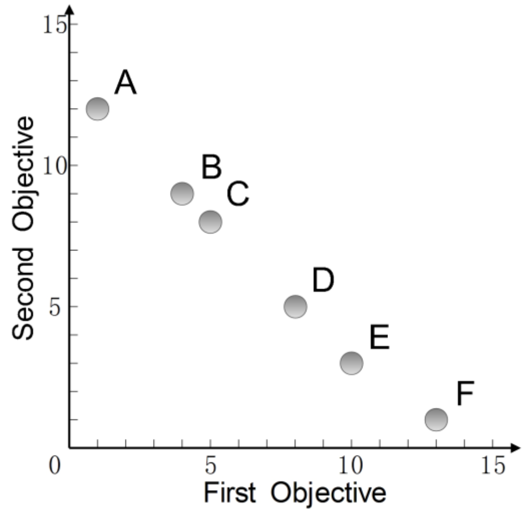


Figure 5.1: An illustrative example where weighted distance may be advantageous over crowding distance.

spaces [77, 106, 141]. The crowding distance operator breaks down as the number of objectives increases due to a bias towards dominance-resistant solutions [77, 91, 105]. Dominance-resistant solutions are undesirable because these non-dominated solutions appear to be desirable, but upon closer examination translate to poor objective function values for the majority of the objectives. When solving MaOPs using the Pareto-dominance relation as the only convergence-related mechanism and the crowding distance operator as the only diversity-related mechanism, non-dominated dominance-resistant solutions will guide the search into sub-optimal areas of the search space. This is, unfortunately, often the case for the original MGPSO algorithm.

The KnMGPSO algorithm is the same as the original MGPSO algorithm except for the differences discussed next. The KnMGPSO algorithm, another adaptation of the original MGPSO algorithm, utilizes knee-points as an additional convergence-related metric to help distinguish desirable solutions for MaOPs. The KnMGPSO identifies a knee-point as follows: for each particle $\mathbf{x}$, $\mathbf{x}$ is considered to be a knee-point if, and only if, $\mathbf{x}$ possesses the maximum objective space distance to the extremal hyperplane H within its neighbourhood (refer to section 3.1.2.1).

Note again that solutions identified as knee-points during the search are often not knee-points of the true POF, since the current found front (i.e. the archive) is simply an approximation of the true POF. Instead, knee-points of the current archive represent particles that are converging best within their neighbourhood and are therefore useful for increasing the selection pressure to converge towards the POF [105]. The KnMGPSO algorithm also aims to find a set of diverse solutions by using the same diversity management and preservation approach already utilized by the original MGPSO; that is, the crowding distance operator. Diversity is further promoted by the knee-points approach that is used since the use of adaptive neighbourhoods (refer to section 3.1.2.1) result in exactly one knee-point per neighbourhood.

Instead of only using crowding distance to determine the archive guide, the KnMGPSO algorithm uses the knee-points identification approach presented in section 3.1.2.1 instead. More specifically, in line 19 of Algorithm 8, the KnMGPSO algorithm first uses knee-points, instead of crowding distance, to determine the winner of the tournament (i.e. the archive guide). If the knee-point identification approach fails to distinguish between solutions, the archive guide is determined as in the original MGPSO algorithm. That is, the least crowded solution of the current tournament will be chosen as the archive guide. By doing this, the KnMGPSO algorithm essentially guides the search around the best converging non-dominated solutions while also taking diversity into account. A tournament size of three, identical to that of the MGPSO algorithm and the PMGPSO algorithm, is used for the archive guide selection. Note that the KnMGPSO algorithm, like the original MGPSO algorithm, also removes dominated solutions from the archive after inserting a new non-dominated solution (lines 4 and 9 of Algorithm 7).

5.3 Empirical Process

Note that this chapter uses the same empirical proces as in section 4.3. Section 5.3.1 discusses the algorithms and parameter tuning approach.

5.3.1 Algorithms and Parameter Tuning Approach

For this part of the study, the following algorithms were investigated:

1. MGPSO$_{STD}$ [53, 136]. Refer to section 3.2.2.2 for detail.

2. MGPSO$_{R}$ [53, 136]. Refer to sections 3.2.2.2 and 3.17 for detail.

3. MGPSO$_{RI}$ [53, 136]. Refer to sections 3.2.2.2 and 3.18 for detail.

4. The knee-point driven multi-guide particle swarm optimization algorithm with the standard static archive balance coefficient update strategy (KnMGPSO$_{STD}$). Refer to sections 3.2.2.2 and 5.2 for detail.

5. The knee-point driven multi-guide particle swarm optimization algorithm with the random dynamic archive balance coefficient update strategy (KnMGPSO$_{R}$). Refer to sections 3.2.2.2 and 5.2 for detail.

6. The knee-point driven multi-guide particle swarm optimization algorithm with the random per particle dynamic archive balance coefficient update strategy (KnMGPSO$_{RI}$). Refer to sections 3.2.2.2 and 5.2 for detail.

7. CDAS-SMPSO [33]. Refer to section 3.2.2.1 for detail.

8. KnEA [174]. Refer to section 3.1.2.1 for detail.

9. MOEA/DD [95]. Refer to section 3.1.2.2 for detail.

10. NSGA-III [38]. Refer to section 3.1.2.3 for detail.

Note that the PMGPSO algorithm was not investigated in this part of the study. The PMGPSO algorithm and the KnMGPSO algorithm are compared in the following chapter.

A summary of the control parameter configurations for the KnMGPSO algorithm is listed below. Note that all of the other control parameter configurations for the KnMGPSO algorithm was kept identical to that of the MGPSO and PMGPSO algorithms (as discussed in section 4.3).

KnMGPSO$_{STD}$, KnMGPSO$_{R}$, KnMGPSO$_{RI}$
The control parameter configurations for the KnMGPSO algorithm were as follows. As the number of objectives increases for MaOPs, Zhang *et al.* [174] suggests increasingly small values for κ, the desired ratio of knee-points to non-dominated solutions. Therefore, κ was set to 0.5 for 3 objectives, 0.4 for 5 objectives, 0.3 for 8 objectives, 0.2 for 10 objectives, and 0.1 for 15 objectives.

Note that the remaining algorithms have already been tuned in Chapter 4. The algorithm control parameter configurations for these algorithms (MGPSO$_{STD}$, MGPSO$_{R}$, MGPSO$_{RI}$, CDAS-SMPSO, KnEA, MOEA/DD, and NSGA-III) can be viewed in section 4.3.2 and Appendix C.

5.4 Results and Discussion

The results of the statistical analysis are shown in tables 5.1 to 5.20, which are then thoroughly analyzed and discussed. The tables contain the overall wins, losses, difference, and rank across the problems for each algorithm. Note that the top-three best overall ranks are highlighted for each table. The HV and IGD performance measure values can be viewed in Appendix E. Sections 5.4.1 and 5.4.2 discuss the findings with respect to HV and IGD respectively. Finally, section 5.4.3 provides some general comments and summarizes the findings.

Table 5.1: HV Ranking for 3-objective DTLZ

Algorithm	**Result**	**3-objective DTLZ 1**	**2**	**3**	**4**	**5**	**6**	**7**	**Overall**
$MGPSO_{STD}$	Wins	7	4	7	0	5	0	4	27
	Losses	0	2	0	0	3	6	0	11
	Difference	7	2	7	0	2	-6	4	16
	Rank	1	6	1	3	5	9	1	1
$MGPSO_{R}$	Wins	1	5	1	0	7	2	4	20
	Losses	5	0	4	0	0	4	0	13
	Difference	-4	5	-3	0	7	-2	4	7
	Rank	7	2	5	3	1	5	1	4
$MGPSO_{RI}$	Wins	1	6	1	0	7	2	4	21
	Losses	6	0	4	0	0	4	0	14
	Difference	-5	6	-3	0	7	-2	4	7
	Rank	9	1	5	3	1	5	1	4
$KnMGPSO_{STD}$	Wins	7	4	6	0	4	0	4	25
	Losses	0	1	0	0	5	8	0	14
	Difference	7	3	6	0	-1	-8	4	11
	Rank	1	5	3	3	6	10	1	3
$KnMGPSO_{R}$	Wins	1	4	1	0	5	1	4	16
	Losses	5	0	4	0	2	4	0	15
	Difference	-4	4	-3	0	3	-3	4	1
	Rank	7	3	5	3	4	7	1	7
$KnMGPSO_{RI}$	Wins	2	4	1	0	6	1	4	18
	Losses	5	0	4	0	0	4	0	13
	Difference	-3	4	-3	0	6	-3	4	5
	Rank	6	3	5	3	3	7	1	6

Table 5.1: HV Ranking for 3-objective DTLZ (continue)

Algorithm	Result	3-objective DTLZ 1	2	3	4	5	6	7	Overall
CDAS-SMPSO	Wins	0	0	0	0	0	6	0	6
	Losses	9	7	9	2	7	2	9	45
	Difference	-9	-7	-9	-2	-7	4	-9	-39
	Rank	10	9	10	9	9	3	10	10
KnEA	Wins	7	3	7	2	0	9	2	30
	Losses	0	6	0	0	6	0	6	18
	Difference	7	-3	7	2	-6	9	-4	12
	Rank	1	7	1	1	8	1	7	2
MOEA/DD	Wins	5	1	1	2	2	8	2	21
	Losses	3	7	4	0	6	1	6	27
	Difference	2	-6	-3	2	-4	7	-4	-6
	Rank	4	8	5	1	7	2	7	8
NSGA-III	Wins	5	0	6	0	0	6	1	18
	Losses	3	8	2	2	7	2	8	32
	Difference	2	-8	4	-2	-7	4	-7	-14
	Rank	4	10	4	9	9	3	9	9

Table 5.2: HV Ranking for 5-objective DTLZ

Algorithm	Result	5-objective DTLZ 1	2	3	4	5	6	7	Overall
$MGPSO_{STD}$	Wins	4	1	5	1	4	1	4	20
	Losses	2	1	2	4	0	2	0	11
	Difference	2	0	3	-3	4	-1	4	9
	Rank	3	6	3	9	2	3	1	2

Table 5.2: HV Ranking for 5-objective DTLZ (continue)

Algorithm	Result	5-objective DTLZ 1	2	3	4	5	6	7	Overall
MGPSO$_R$	Wins	2	2	1	1	4	1	4	15
	Losses	3	1	4	1	0	2	0	11
	Difference	-1	1	-3	0	4	-1	4	4
	Rank	7	2	8	5	2	3	1	7
MGPSO$_{RI}$	Wins	2	2	1	2	5	1	4	17
	Losses	2	1	4	1	0	2	0	10
	Difference	0	1	-3	1	5	-1	4	7
	Rank	5	2	8	2	1	3	1	5
KnMGPSO$_{STD}$	Wins	3	1	3	1	4	1	4	17
	Losses	2	1	2	1	0	2	0	8
	Difference	1	0	1	0	4	-1	4	9
	Rank	4	6	4	5	2	3	1	2
KnMGPSO$_R$	Wins	2	2	1	2	4	1	4	16
	Losses	2	1	3	1	0	2	0	9
	Difference	0	1	-2	1	4	-1	4	7
	Rank	5	2	5	2	2	3	1	5
KnMGPSO$_{RI}$	Wins	2	2	1	1	4	1	4	15
	Losses	4	1	3	1	1	2	0	12
	Difference	-2	1	-2	0	3	-1	4	3
	Rank	8	2	5	5	6	3	1	8
CDAS-SMPSO	Wins	1	0	0	1	1	1	0	4
	Losses	8	8	8	1	7	2	9	43
	Difference	-7	-8	-8	0	-6	-1	-9	-39
	Rank	9	10	10	5	8	3	10	9

Table 5.2: HV Ranking for 5-objective DTLZ (continue)

Algorithm	**Result**	**5-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
KnEA	Wins	0	0	0	0	0	0	2	2
	Losses	9	1	2	9	9	9	6	45
	Difference	-9	-1	-2	-9	-9	-9	-4	-43
	Rank	10	8	5	10	10	10	7	10
MOEA/DD	Wins	8	9	8	9	3	9	1	47
	Losses	0	0	1	0	6	0	6	13
	Difference	8	9	7	9	-3	9	-5	34
	Rank	1	1	2	1	7	1	8	1
NSGA-III	Wins	8	1	9	2	1	8	1	30
	Losses	0	5	0	1	7	1	7	21
	Difference	8	-4	9	1	-6	7	-6	9
	Rank	1	9	1	2	8	2	9	2

Table 5.3: HV Ranking for 8-objective DTLZ

Algorithm	**Result**	**8-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
MGPSO_{STD}	Wins	5	3	4	6	5	2	8	33
	Losses	3	0	2	3	0	3	0	11
	Difference	2	3	2	3	5	-1	8	22
	Rank	4	2	3	4	1	4	1	2
MGPSO_{R}	Wins	2	3	1	1	5	1	4	17
	Losses	4	0	3	5	0	3	2	17
	Difference	-2	3	-2	-4	5	-2	2	0
	Rank	6	2	6	6	1	5	3	5

Table 5.3: HV Ranking for 8-objective DTLZ (continue)

Algorithm	**Result**	**8-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$MGPSO_{RI}$	Wins	1	4	1	1	4	1	4	16
	Losses	4	0	2	5	0	3	2	16
	Difference	-3	4	-1	-4	4	-2	2	0
	Rank	7	1	4	6	4	5	3	5
$KnMGPSO_{STD}$	Wins	2	3	1	5	5	1	8	25
	Losses	3	0	2	4	0	3	0	12
	Difference	-1	3	-1	1	5	-2	8	13
	Rank	5	2	4	5	1	5	1	4
$KnMGPSO_{R}$	Wins	1	3	1	1	4	1	4	15
	Losses	4	1	3	5	3	4	2	22
	Difference	-3	2	-2	-4	1	-3	2	-7
	Rank	7	7	6	6	6	9	3	8
$KnMGPSO_{RI}$	Wins	1	3	1	1	4	1	4	15
	Losses	6	0	3	5	0	3	2	19
	Difference	-5	3	-2	-4	4	-2	2	-4
	Rank	9	2	6	6	4	5	3	7
CDAS-SMPSO	Wins	0	1	0	7	1	7	0	16
	Losses	9	8	8	2	6	1	9	43
	Difference	-9	-7	-8	5	-5	6	-9	-27
	Rank	10	9	10	3	7	2	10	9
KnEA	Wins	8	0	0	0	0	0	2	10
	Losses	0	9	2	9	9	9	6	44
	Difference	8	-9	-2	-9	-9	-9	-4	-34
	Rank	1	10	6	10	10	10	7	10

Table 5.3: HV Ranking for 8-objective DTLZ (continue)

Algorithm	**Result**	**8-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
MOEA/DD	Wins	7	3	8	9	1	9	1	38
	Losses	1	0	0	0	6	0	8	15
	Difference	6	3	8	9	-5	9	-7	23
	Rank	3	2	1	1	7	1	9	1
NSGA-III	Wins	7	2	8	8	1	7	2	35
	Losses	0	7	0	1	6	1	6	21
	Difference	7	-5	8	7	-5	6	-4	14
	Rank	2	8	1	2	7	2	7	3

Table 5.4: HV Ranking for 10-objective DTLZ

Algorithm	**Result**	**10-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$MGPSO_{STD}$	Wins	1	4	0	5	4	4	8	26
	Losses	1	0	2	3	0	1	0	7
	Difference	0	4	-2	2	4	3	8	19
	Rank	2	2	5	5	1	2	1	1
$MGPSO_R$	Wins	1	4	0	0	4	2	4	15
	Losses	1	1	2	5	0	1	2	12
	Difference	0	3	-2	-5	4	1	2	3
	Rank	2	3	5	6	1	3	3	4
$MGPSO_{RI}$	Wins	1	8	0	0	4	2	4	19
	Losses	1	0	2	5	0	1	2	11
	Difference	0	8	-2	-5	4	1	2	8
	Rank	2	1	5	6	1	3	3	3

Table 5.4: HV Ranking for 10-objective DTLZ (continue)

Algorithm	Result	1	2	3	4	5	6	7	Overall
$KnMGPSO_{STD}$	Wins	1	4	0	5	4	2	8	24
	Losses	1	1	2	2	0	1	0	7
	Difference	0	3	-2	3	4	1	8	17
	Rank	2	3	5	4	1	3	1	2
$KnMGPSO_{R}$	Wins	1	4	0	0	4	2	4	15
	Losses	1	1	2	5	0	3	2	14
	Difference	0	3	-2	-5	4	-1	2	1
	Rank	2	3	5	6	1	8	3	6
$KnMGPSO_{RI}$	Wins	1	4	0	0	4	2	4	15
	Losses	1	1	2	5	0	2	2	13
	Difference	0	3	-2	-5	4	0	2	2
	Rank	2	3	5	6	1	7	3	5
CDAS-SMPSO	Wins	0	1	0	6	3	9	0	19
	Losses	9	6	1	0	6	0	9	31
	Difference	-9	-5	-1	6	-3	9	-9	-12
	Rank	10	7	3	3	7	1	10	8
KnEA	Wins	9	1	0	0	1	0	2	13
	Losses	0	6	1	5	7	8	7	34
	Difference	9	-5	-1	-5	-6	-8	-5	-21
	Rank	1	7	3	6	8	10	8	10
MOEA/DD	Wins	1	0	6	7	1	0	1	16
	Losses	1	9	0	0	7	7	8	32
	Difference	0	-9	6	7	-6	-7	-7	-16
	Rank	2	10	2	1	8	9	9	9

Table 5.4: HV Ranking for 10-objective DTLZ (continue)

Algorithm	**Result**	**10-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
NSGA-III	Wins	1	1	8	7	0	2	3	22
	Losses	1	6	0	0	9	1	6	23
	Difference	0	-5	8	7	-9	1	-3	-1
	Rank	2	7	1	1	10	3	7	7

Table 5.5: HV Ranking for 15-objective DTLZ

Algorithm	**Result**	**15-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
MGPSO_{STD}	Wins	2	4	1	5	4	0	8	24
	Losses	2	0	2	0	0	1	0	5
	Difference	0	4	-1	5	4	-1	8	19
	Rank	3	1	4	1	1	5	1	1
MGPSO_{R}	Wins	1	4	0	0	4	1	4	14
	Losses	2	0	3	2	0	1	2	10
	Difference	-1	4	-3	-2	4	0	2	4
	Rank	4	1	6	8	1	2	3	4
MGPSO_{RI}	Wins	1	4	1	0	4	0	4	14
	Losses	2	0	2	2	0	1	2	9
	Difference	-1	4	-1	-2	4	-1	2	5
	Rank	4	1	4	8	1	5	3	3
KnMGPSO_{STD}	Wins	1	4	0	5	4	1	8	23
	Losses	2	0	3	0	0	1	0	6
	Difference	-1	4	-3	5	4	0	8	17
	Rank	4	1	6	1	1	2	1	2

Table 5.5: HV Ranking for 15-objective DTLZ (continue)

Algorithm	Result	15-objective DTLZ 1	2	3	4	5	6	7	Overall
KnMGPSO_R	Wins	1	4	0	1	4	0	4	14
	Losses	2	0	3	2	0	1	2	10
	Difference	-1	4	-3	-1	4	-1	2	4
	Rank	4	1	6	6	1	5	3	4
KnMGPSO_{RI}	Wins	1	4	0	1	4	0	4	14
	Losses	2	0	5	2	0	1	2	12
	Difference	-1	4	-5	-1	4	-1	2	2
	Rank	4	1	10	6	1	5	3	6
CDAS-SMPSO	Wins	0	3	0	0	3	8	0	14
	Losses	8	6	3	0	6	0	9	32
	Difference	-8	-3	-3	0	-3	8	-9	-18
	Rank	10	7	6	3	7	1	10	10
KnEA	Wins	8	0	5	0	1	0	2	16
	Losses	0	7	0	4	7	1	7	26
	Difference	8	-7	5	-4	-6	-1	-5	-10
	Rank	1	9	3	10	8	5	8	8
MOEA/DD	Wins	8	0	7	0	1	0	1	17
	Losses	0	8	0	0	7	0	8	23
	Difference	8	-8	7	0	-6	0	-7	-6
	Rank	1	10	1	3	8	2	9	7
NSGA-III	Wins	0	1	7	0	0	0	3	11
	Losses	3	7	0	0	9	3	6	28
	Difference	-3	-6	7	0	-9	-3	-3	-17
	Rank	9	8	1	3	10	10	7	9

Table 5.6: HV Ranking for 3-objective WFG

Algorithm	**Result**	**3-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
MGPSO$_{STD}$	Wins	0	0	3	2	0	2	0	0	4	11
	Losses	8	5	4	3	1	5	4	5	1	36
	Difference	-8	-5	-1	-1	-1	-3	-4	-5	3	-25
	Rank	9	9	6	6	4	7	5	10	5	9
MGPSO$_{R}$	Wins	2	0	5	2	0	5	0	1	4	19
	Losses	4	3	0	3	1	1	4	4	0	20
	Difference	-2	-3	5	-1	-1	4	-4	-3	4	-1
	Rank	5	6	4	6	4	2	5	5	4	6
MGPSO$_{RI}$	Wins	2	2	6	2	0	5	0	0	5	22
	Losses	4	3	0	2	1	1	4	4	0	19
	Difference	-2	-1	6	0	-1	4	-4	-4	5	3
	Rank	5	4	1	5	4	2	5	6	2	5
KnMGPSO$_{STD}$	Wins	0	0	4	2	0	2	0	0	4	12
	Losses	8	5	3	1	3	6	4	4	3	37
	Difference	-8	-5	1	1	-3	-4	-4	-4	1	-25
	Rank	9	9	5	4	8	8	5	6	6	9
KnMGPSO$_{R}$	Wins	2	0	6	5	0	5	0	0	6	24
	Losses	4	3	0	1	3	1	4	4	0	20
	Difference	-2	-3	6	4	-3	4	-4	-4	6	4
	Rank	5	6	1	3	8	2	5	6	1	4
KnMGPSO$_{RI}$	Wins	2	2	6	6	0	5	0	0	5	26
	Losses	4	3	0	0	4	1	4	4	0	20
	Difference	-2	-1	6	6	-4	4	-4	-4	5	6
	Rank	5	4	1	2	10	2	5	6	2	3

Table 5.6: HV Ranking for 3-objective WFG (continue)

Algorithm	**Result**	**3-objective WFG 1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Overall**
CDAS-SMPSO	Wins	7	8	3	2	4	3	8	8	3	46
	Losses	0	0	5	3	1	5	1	1	6	22
	Difference	7	8	-2	-1	3	-2	7	7	-3	24
	Rank	1	1	7	6	2	6	2	2	7	2
KnEA	Wins	6	8	1	8	9	9	9	9	2	61
	Losses	3	0	8	0	0	0	0	0	7	18
	Difference	3	8	-7	8	9	9	9	9	-5	43
	Rank	4	1	9	1	1	1	1	1	8	1
MOEA/DD	Wins	7	7	0	0	1	0	6	6	0	27
	Losses	0	2	9	9	3	8	3	3	8	45
	Difference	7	5	-9	-9	-2	-8	3	3	-8	-18
	Rank	1	3	10	10	7	9	4	4	9	8
NSGA-III	Wins	7	0	2	1	4	0	7	7	0	28
	Losses	0	3	7	8	1	8	2	2	8	39
	Difference	7	-3	-5	-7	3	-8	5	5	-8	-11
	Rank	1	6	8	9	2	9	3	3	9	7

Table 5.7: HV Ranking for 5-objective WFG

Algorithm	**Result**	**5-objective WFG 1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Overall**
$MGPSO_{STD}$	Wins	0	1	4	0	2	0	0	0	4	11
	Losses	8	2	4	6	4	7	4	4	0	39
	Difference	-8	-1	0	-6	-2	-7	-4	-4	4	-28
	Rank	9	7	5	10	6	9	5	5	1	9

Table 5.7: HV Ranking for 5-objective WFG (continue)

Algorithm	Result	5-objective WFG 1	2	3	4	5	6	7	8	9	Overall
$MGPSO_R$	Wins	2	2	6	0	3	2	0	0	4	19
	Losses	4	1	0	5	4	3	4	4	0	25
	Difference	-2	1	6	-5	-1	-1	-4	-4	4	-6
	Rank	5	3	1	8	5	5	5	5	1	5
$MGPSO_{RI}$	Wins	2	1	6	0	2	2	0	0	4	17
	Losses	4	1	0	5	4	3	4	4	0	25
	Difference	-2	0	6	-5	-2	-1	-4	-4	4	-8
	Rank	5	6	1	8	6	5	5	5	1	7
$KnMGPSO_{STD}$	Wins	0	1	4	0	0	0	0	0	4	9
	Losses	8	2	4	4	5	8	4	4	0	39
	Difference	-8	-1	0	-4	-5	-8	-4	-4	4	-30
	Rank	9	7	5	7	8	10	5	5	1	10
$KnMGPSO_R$	Wins	2	4	6	3	0	2	0	0	4	21
	Losses	4	1	0	3	7	4	4	4	0	27
	Difference	-2	3	6	0	-7	-2	-4	-4	4	-6
	Rank	5	2	1	5	9	7	5	5	1	5
$KnMGPSO_{RI}$	Wins	2	2	6	1	0	1	0	0	4	16
	Losses	4	1	0	4	7	4	4	4	0	28
	Difference	-2	1	6	-3	-7	-3	-4	-4	4	-12
	Rank	5	3	1	6	9	8	5	5	1	8
CDAS-SMPSO	Wins	7	8	3	7	6	4	6	6	2	49
	Losses	0	0	6	1	3	2	1	3	6	22
	Difference	7	8	-3	6	3	2	5	3	-4	27
	Rank	1	1	7	2	4	4	2	4	7	2

Table 5.7: HV Ranking for 5-objective WFG (continue)

Algorithm	Result	5-objective WFG									Overall
		1	2	3	4	5	6	7	8	9	
KnEA	Wins	6	1	1	9	9	9	9	9	2	55
	Losses	3	0	8	0	0	0	0	0	6	17
	Difference	3	1	-7	9	9	9	9	9	-4	38
	Rank	4	3	9	1	1	1	1	1	7	1
MOEA/DD	Wins	7	0	0	7	7	7	6	7	0	41
	Losses	0	8	9	1	1	1	1	1	8	30
	Difference	7	-8	-9	6	6	6	5	6	-8	11
	Rank	1	10	10	2	2	2	2	2	9	4
NSGA-III	Wins	7	0	2	5	7	6	6	7	0	40
	Losses	0	4	7	3	1	1	1	1	8	26
	Difference	7	-4	-5	2	6	5	5	6	-8	14
	Rank	1	9	8	4	2	3	2	2	9	3

Table 5.8: HV Ranking for 8-objective WFG

Algorithm	Result	8-objective WFG									Overall
		1	2	3	4	5	6	7	8	9	
$MGPSO_{STD}$	Wins	0	4	4	3	5	4	1	1	4	26
	Losses	4	0	0	0	2	0	3	3	0	12
	Difference	-4	4	4	3	3	4	-2	-2	4	14
	Rank	7	1	2	2	4	3	5	5	1	4
$MGPSO_{R}$	Wins	4	4	4	3	2	5	1	1	4	28
	Losses	0	0	1	0	5	0	4	3	0	13
	Difference	4	4	3	3	-3	5	-3	-2	4	15
	Rank	1	1	3	2	7	1	7	5	1	3

Table 5.8: HV Ranking for 8-objective WFG (continue)

Algorithm	Result	1	2	3	4	5	6	7	8	9	Overall
		8-objective WFG									
$MGPSO_{RI}$	Wins	4	4	4	3	3	5	1	1	4	29
	Losses	0	0	1	0	4	0	4	3	0	12
	Difference	4	4	3	3	-1	5	-3	-2	4	17
	Rank	1	1	3	2	6	1	7	5	1	2
$KnMGPSO_{STD}$	Wins	0	4	8	3	4	4	4	3	4	34
	Losses	4	0	0	0	3	2	2	3	0	14
	Difference	-4	4	8	3	1	2	2	0	4	20
	Rank	7	1	1	2	5	6	4	4	1	1
$KnMGPSO_{R}$	Wins	4	4	4	4	1	4	1	1	4	27
	Losses	0	0	1	0	6	0	4	4	0	15
	Difference	4	4	3	4	-5	4	-3	-3	4	12
	Rank	1	1	3	1	8	3	7	8	1	5
$KnMGPSO_{RI}$	Wins	4	4	4	3	1	4	1	1	4	26
	Losses	0	0	1	0	7	0	3	4	0	15
	Difference	4	4	3	3	-6	4	-2	-3	4	11
	Rank	1	1	3	2	9	3	5	8	1	6
CDAS-SMPSO	Wins	0	1	3	3	9	1	8	7	2	34
	Losses	4	6	6	1	0	6	0	0	6	29
	Difference	-4	-5	-3	2	9	-5	8	7	-4	5
	Rank	7	7	7	7	1	7	1	2	7	7
KnEA	Wins	0	1	0	0	0	0	0	0	0	1
	Losses	0	6	8	9	9	9	9	9	9	68
	Difference	0	-5	-8	-9	-9	-9	-9	-9	-9	-67
	Rank	5	7	9	10	10	10	10	10	10	10

Table 5.8: HV Ranking for 8-objective WFG (continue)

Algorithm	**Result**	**8-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
MOEA/DD	Wins	0	0	0	1	6	1	6	7	1	22
	Losses	4	9	8	7	2	6	2	1	8	47
	Difference	-4	-9	-8	-6	4	-5	4	6	-7	-25
	Rank	7	10	9	8	3	7	3	3	9	9
NSGA-III	Wins	0	1	2	1	8	1	8	8	2	31
	Losses	0	6	7	7	1	6	0	0	6	33
	Difference	0	-5	-5	-6	7	-5	8	8	-4	-2
	Rank	5	7	8	8	2	7	1	1	7	8

Table 5.9: HV Ranking for 10-objective WFG

Algorithm	**Result**	**10-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
$MGPSO_{STD}$	Wins	0	4	4	4	7	4	2	7	4	36
	Losses	5	0	1	2	0	1	0	0	0	9
	Difference	-5	4	3	2	7	3	2	7	4	27
	Rank	9	1	3	5	1	6	2	1	1	6
$MGPSO_{R}$	Wins	5	4	4	5	2	4	2	4	4	34
	Losses	0	0	2	0	2	0	0	1	0	5
	Difference	5	4	2	5	0	4	2	3	4	29
	Rank	1	1	6	3	4	2	2	4	1	3
$MGPSO_{RI}$	Wins	3	4	4	6	2	5	2	3	4	33
	Losses	0	0	1	0	2	0	0	1	0	4
	Difference	3	4	3	6	0	5	2	2	4	29
	Rank	3	1	3	1	4	1	2	5	1	3

Table 5.9: HV Ranking for 10-objective WFG (continue)

Algorithm	Result	1	2	3	4	5	6	7	8	9	Overall
$KnMGPSO_{STD}$	Wins	1	4	8	3	7	4	3	4	4	38
	Losses	4	0	0	4	0	0	0	0	0	8
	Difference	-3	4	8	-1	7	4	3	4	4	30
	Rank	8	1	1	6	1	2	1	2	1	1
$KnMGPSO_{R}$	Wins	4	4	4	6	2	4	2	3	4	33
	Losses	0	0	1	0	2	0	0	1	0	4
	Difference	4	4	3	6	0	4	2	2	4	29
	Rank	2	1	3	1	4	2	2	5	1	3
$KnMGPSO_{RI}$	Wins	3	4	5	5	2	4	2	4	4	33
	Losses	0	0	0	0	2	0	1	0	0	3
	Difference	3	4	5	5	0	4	1	4	4	30
	Rank	3	1	2	3	4	2	8	2	1	1
CDAS-SMPSO	Wins	0	1	3	3	2	2	2	2	0	15
	Losses	7	8	6	5	2	6	0	4	6	44
	Difference	-7	-7	-3	-2	0	-4	2	-2	-6	-29
	Rank	10	9	7	7	4	7	2	7	7	7
KnEA	Wins	2	2	1	0	2	2	0	0	0	9
	Losses	0	6	8	7	0	6	8	8	6	49
	Difference	2	-4	-7	-7	2	-4	-8	-8	-6	-40
	Rank	5	7	9	8	3	7	9	9	7	8
MOEA/DD	Wins	1	0	0	0	0	0	0	0	0	1
	Losses	1	9	9	7	9	9	8	8	6	66
	Difference	0	-9	-9	-7	-9	-9	-8	-8	-6	-65
	Rank	6	10	10	8	10	10	9	9	7	10

Table 5.9: HV Ranking for 10-objective WFG (continue)

Algorithm	Result	10-objective WFG 1	2	3	4	5	6	7	8	9	Overall
NSGA-III	Wins	0	2	2	0	1	1	2	2	0	10
	Losses	2	6	7	7	8	8	0	6	6	50
	Difference	-2	-4	-5	-7	-7	-7	2	-4	-6	-40
	Rank	7	7	8	8	9	9	2	8	7	8

Table 5.10: HV Ranking for 15-objective WFG

Algorithm	Result	15-objective WFG 1	2	3	4	5	6	7	8	9	Overall
MGPSO_{STD}	Wins	1	3	4	2	6	3	1	1	2	23
	Losses	0	0	0	0	1	3	2	1	0	7
	Difference	1	3	4	2	5	0	-1	0	2	16
	Rank	1	1	1	3	2	7	4	3	2	3
MGPSO_{R}	Wins	1	3	4	3	2	4	1	1	3	22
	Losses	0	0	0	0	3	1	2	1	0	7
	Difference	1	3	4	3	-1	3	-1	0	3	15
	Rank	1	1	1	1	5	2	4	3	1	4
MGPSO_{RI}	Wins	1	3	4	3	2	3	1	1	2	20
	Losses	0	0	0	0	3	1	2	1	0	7
	Difference	1	3	4	3	-1	2	-1	0	2	13
	Rank	1	1	1	1	5	4	4	3	2	5
KnMGPSO_{STD}	Wins	1	3	4	2	6	3	1	1	2	23
	Losses	0	0	0	0	1	1	2	1	0	5
	Difference	1	3	4	2	5	2	-1	0	2	18
	Rank	1	1	1	3	2	4	4	3	2	2

Table 5.10: HV Ranking for 15-objective WFG (continue)

Algorithm	**Result**	**15-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
KnMGPSO$_R$	Wins	1	3	4	2	2	4	1	1	2	20
	Losses	0	0	0	0	3	1	2	1	0	7
	Difference	1	3	4	2	-1	3	-1	0	2	13
	Rank	1	1	1	3	5	2	4	3	2	5
KnMGPSO$_{RI}$	Wins	1	3	4	2	2	3	1	1	2	19
	Losses	0	0	0	0	3	1	2	1	0	7
	Difference	1	3	4	2	-1	2	-1	0	2	12
	Rank	1	1	1	3	5	4	4	3	2	7
CDAS-SMPSO	Wins	0	0	3	1	0	1	1	1	1	8
	Losses	0	7	6	8	8	7	1	1	8	46
	Difference	0	-7	-3	-7	-8	-6	0	0	-7	-38
	Rank	7	9	7	9	9	8	3	3	9	9
KnEA	Wins	0	1	0	0	0	0	0	0	0	1
	Losses	6	7	8	9	8	9	9	9	9	74
	Difference	-6	-6	-8	-9	-8	-9	-9	-9	-9	-73
	Rank	10	8	9	10	9	10	10	10	10	10
MOEA/DD	Wins	0	0	0	2	2	1	7	1	2	15
	Losses	0	8	8	0	1	7	0	0	1	25
	Difference	0	-8	-8	2	1	-6	7	1	1	-10
	Rank	7	10	9	3	4	8	2	2	8	8
NSGA-III	Wins	0	3	2	2	9	9	8	8	2	43
	Losses	0	0	7	2	0	0	0	0	0	9
	Difference	0	3	-5	0	9	9	8	8	2	34
	Rank	7	1	8	8	1	1	1	1	2	1

Table 5.11: IGD Ranking for 3-objective DTLZ

Algorithm	**Result**	**3-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$MGPSO_{STD}$	Wins	1	1	2	0	0	4	3	11
	Losses	7	3	5	4	3	0	4	26
	Difference	-6	-2	-3	-4	-3	4	-1	-15
	Rank	8	5	6	8	6	2	5	9
$MGPSO_{R}$	Wins	4	0	6	0	0	4	0	14
	Losses	2	4	0	4	5	0	6	21
	Difference	2	-4	6	-4	-5	4	-6	-7
	Rank	3	8	1	8	9	2	8	7
$MGPSO_{RI}$	Wins	4	0	6	0	0	4	0	14
	Losses	2	4	0	4	3	0	6	19
	Difference	2	-4	6	-4	-3	4	-6	-5
	Rank	3	8	1	8	6	2	8	5
$KnMGPSO_{STD}$	Wins	1	0	1	0	2	5	3	12
	Losses	7	3	5	3	3	0	4	25
	Difference	-6	-3	-4	-3	-1	5	-1	-13
	Rank	8	6	8	5	4	1	5	8
$KnMGPSO_{R}$	Wins	4	0	6	0	0	4	0	14
	Losses	2	3	0	3	3	0	6	17
	Difference	2	-3	6	-3	-3	4	-6	-3
	Rank	3	6	1	5	6	2	8	4
$KnMGPSO_{RI}$	Wins	4	0	6	0	0	4	0	14
	Losses	2	5	0	3	5	1	4	20
	Difference	2	-5	6	-3	-5	3	-4	-6
	Rank	3	10	1	5	9	6	7	6

Table 5.11: IGD Ranking for 3-objective DTLZ (continue)

Algorithm	Result	3-objective DTLZ 1	2	3	4	5	6	7	Overall
CDAS-SMPSO	Wins	0	3	0	3	2	0	6	14
	Losses	9	3	9	3	3	9	3	39
	Difference	-9	0	-9	0	-1	-9	3	-25
	Rank	10	4	10	4	4	10	4	10
KnEA	Wins	3	7	1	7	7	1	8	34
	Losses	6	2	7	1	2	8	1	27
	Difference	-3	5	-6	6	5	-7	7	7
	Rank	7	3	9	2	3	9	2	3
MOEA/DD	Wins	9	9	5	9	9	3	7	51
	Losses	0	0	4	0	0	6	2	12
	Difference	9	9	1	9	9	-3	5	39
	Rank	1	1	5	1	1	7	3	1
NSGA-III	Wins	8	8	2	7	8	2	9	44
	Losses	1	1	5	1	1	7	0	16
	Difference	7	7	-3	6	7	-5	9	28
	Rank	2	2	6	2	2	8	1	2

Table 5.12: IGD Ranking for 5-objective DTLZ

Algorithm	Result	5-objective DTLZ 1	2	3	4	5	6	7	Overall
$MGPSO_{STD}$	Wins	3	3	4	0	2	4	0	16
	Losses	2	2	4	4	2	0	5	19
	Difference	1	1	0	-4	0	4	-5	-3
	Rank	3	3	6	10	3	1	9	8

Table 5.12: IGD Ranking for 5-objective DTLZ (continue)

Algorithm	**Result**	**5-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
MGPSO_R	Wins	1	2	5	0	2	4	1	15
	Losses	3	2	0	2	2	0	4	13
	Difference	-2	0	5	-2	0	4	-3	2
	Rank	8	6	1	5	3	1	6	6
MGPSO_{RI}	Wins	1	2	5	0	2	4	0	14
	Losses	2	5	0	2	2	0	4	15
	Difference	-1	-3	5	-2	0	4	-4	-1
	Rank	4	8	1	5	3	1	8	7
KnMGPSO_{STD}	Wins	1	3	4	0	2	4	1	15
	Losses	2	2	0	2	2	0	4	12
	Difference	-1	1	4	-2	0	4	-3	3
	Rank	4	3	5	5	3	1	6	5
KnMGPSO_R	Wins	1	2	5	1	2	4	6	21
	Losses	3	2	0	2	2	0	2	11
	Difference	-2	0	5	-1	0	4	4	10
	Rank	8	6	1	3	3	1	3	4
KnMGPSO_{RI}	Wins	1	3	5	0	2	4	6	21
	Losses	2	2	0	2	2	0	2	10
	Difference	-1	1	5	-2	0	4	4	11
	Rank	4	3	1	5	3	1	3	3
CDAS-SMPSO	Wins	1	0	0	1	1	0	0	3
	Losses	2	9	9	2	8	8	7	45
	Difference	-1	-9	-9	-1	-7	-8	-7	-42
	Rank	4	10	10	3	9	9	10	10

Table 5.12: IGD Ranking for 5-objective DTLZ (continue)

Algorithm	**Result**	**5-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
KnEA	Wins	0	1	1	0	0	0	8	10
	Losses	9	8	8	2	9	8	1	45
	Difference	-9	-7	-7	-2	-9	-8	7	-35
	Rank	10	9	9	5	10	9	2	9
MOEA/DD	Wins	9	8	2	8	8	3	2	40
	Losses	0	1	7	0	1	6	4	19
	Difference	9	7	-5	8	7	-3	-2	21
	Rank	1	2	8	1	2	7	5	2
NSGA-III	Wins	8	9	3	8	9	2	9	48
	Losses	1	0	6	0	0	7	0	14
	Difference	7	9	-3	8	9	-5	9	34
	Rank	2	1	7	1	1	8	1	1

Table 5.13: IGD Ranking for 8-objective DTLZ

Algorithm	**Result**	**8-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$MGPSO_{STD}$	Wins	0	2	4	5	2	4	2	19
	Losses	4	1	3	3	2	0	5	18
	Difference	-4	1	1	2	0	4	-3	1
	Rank	5	5	6	4	3	1	7	8
$MGPSO_{R}$	Wins	0	2	4	1	2	4	6	19
	Losses	4	2	1	4	2	0	1	14
	Difference	-4	0	3	-3	0	4	5	5
	Rank	5	6	4	6	3	1	2	5

Table 5.13: IGD Ranking for 8-objective DTLZ (continue)

Algorithm	Result	8-objective DTLZ 1	2	3	4	5	6	7	Overall
$MGPSO_{RI}$	Wins	0	2	4	1	2	4	6	19
	Losses	4	3	1	4	2	0	1	15
	Difference	-4	-1	3	-3	0	4	5	4
	Rank	5	7	4	6	3	1	2	7
$KnMGPSO_{STD}$	Wins	0	6	5	3	2	4	2	22
	Losses	4	0	0	3	2	0	6	15
	Difference	-4	6	5	0	0	4	-4	7
	Rank	5	1	2	5	3	1	8	3
$KnMGPSO_{R}$	Wins	0	5	7	1	2	4	3	22
	Losses	4	0	0	5	2	0	4	15
	Difference	-4	5	7	-4	0	4	-1	7
	Rank	5	2	1	8	3	1	6	3
$KnMGPSO_{RI}$	Wins	0	4	5	1	2	4	4	20
	Losses	4	0	0	5	2	0	4	15
	Difference	-4	4	5	-4	0	4	0	5
	Rank	5	3	2	8	3	1	5	5
CDAS-SMPSO	Wins	6	1	0	7	1	0	0	15
	Losses	2	8	9	2	8	8	9	46
	Difference	4	-7	-9	5	-7	-8	-9	-31
	Rank	3	9	10	3	9	9	10	10
KnEA	Wins	8	0	1	0	0	0	6	15
	Losses	1	9	8	9	9	8	1	45
	Difference	7	-9	-7	-9	-9	-8	5	-30
	Rank	2	10	9	10	10	9	2	9

Table 5.13: IGD Ranking for 8-objective DTLZ (continue)

Algorithm	Result	8-objective DTLZ 1	2	3	4	5	6	7	Overall
MOEA/DD	Wins	9	2	2	8	8	2	1	32
	Losses	0	3	6	0	1	6	8	24
	Difference	9	-1	-4	8	7	-4	-7	8
	Rank	1	7	7	1	2	7	9	2
NSGA-III	Wins	6	2	2	8	9	2	9	38
	Losses	2	0	6	0	0	6	0	14
	Difference	4	2	-4	8	9	-4	9	24
	Rank	3	4	7	1	1	7	1	1

Table 5.14: IGD Ranking for 10-objective DTLZ

Algorithm	Result	10-objective DTLZ 1	2	3	4	5	6	7	Overall
MGPSO_{STD}	Wins	0	4	4	5	4	4	4	25
	Losses	4	0	1	3	0	0	3	11
	Difference	-4	4	3	2	4	4	1	14
	Rank	5	4	4	4	1	1	5	2
MGPSO_{R}	Wins	0	4	4	0	4	4	6	22
	Losses	4	0	1	5	0	0	0	10
	Difference	-4	4	3	-5	4	4	6	12
	Rank	5	4	4	6	1	1	2	5
MGPSO_{RI}	Wins	0	4	4	0	4	4	8	24
	Losses	4	3	2	5	0	0	0	14
	Difference	-4	1	2	-5	4	4	8	10
	Rank	5	6	6	6	1	1	1	6

Table 5.14: IGD Ranking for 10-objective DTLZ (continue)

Algorithm	Result	10-objective DTLZ 1	2	3	4	5	6	7	Overall
KnMGPSO_{STD}	Wins	0	5	5	5	4	4	2	25
	Losses	4	0	0	3	0	0	4	11
	Difference	-4	5	5	2	4	4	-2	14
	Rank	5	1	2	4	1	1	6	2
KnMGPSO_{R}	Wins	0	5	7	0	4	4	5	25
	Losses	4	0	0	5	0	0	1	10
	Difference	-4	5	7	-5	4	4	4	15
	Rank	5	1	1	6	1	1	4	1
KnMGPSO_{RI}	Wins	0	5	4	0	4	4	6	23
	Losses	4	0	0	5	0	0	1	10
	Difference	-4	5	4	-5	4	4	5	13
	Rank	5	1	3	6	1	1	3	4
CDAS-SMPSO	Wins	6	1	0	7	1	0	0	15
	Losses	2	7	9	2	7	8	9	44
	Difference	4	-6	-9	5	-6	-8	-9	-29
	Rank	3	8	10	3	9	10	10	9
KnEA	Wins	8	0	1	0	0	0	2	11
	Losses	0	7	8	5	9	7	5	41
	Difference	8	-7	-7	-5	-9	-7	-3	-30
	Rank	1	9	9	6	10	9	7	10
MOEA/DD	Wins	8	0	2	8	2	1	1	22
	Losses	0	8	7	0	6	6	8	35
	Difference	8	-8	-5	8	-4	-5	-7	-13
	Rank	1	10	8	1	7	8	9	8

Table 5.14: IGD Ranking for 10-objective DTLZ (continue)

Algorithm	**Result**	**10-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
NSGA-III	Wins	6	3	3	8	1	2	2	25
	Losses	2	6	6	0	6	6	5	31
	Difference	4	-3	-3	8	-5	-4	-3	-6
	Rank	3	7	7	1	8	7	7	7

Table 5.15: IGD Ranking for 15-objective DTLZ

Algorithm	**Result**	**15-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
MGPSO_{STD}	Wins	0	3	3	4	4	3	2	19
	Losses	4	2	4	0	0	0	4	14
	Difference	-4	1	-1	4	4	3	-2	5
	Rank	5	6	5	2	1	1	5	6
MGPSO_{R}	Wins	0	3	3	0	4	3	6	19
	Losses	4	1	4	3	0	0	0	12
	Difference	-4	2	-1	-3	4	3	6	7
	Rank	5	4	5	9	1	1	1	5
MGPSO_{RI}	Wins	0	3	3	0	4	3	6	19
	Losses	4	1	4	2	0	0	0	11
	Difference	-4	2	-1	-2	4	3	6	8
	Rank	5	4	5	8	1	1	1	4
KnMGPSO_{STD}	Wins	0	4	6	5	4	3	2	24
	Losses	4	0	1	0	0	0	4	9
	Difference	-4	4	5	5	4	3	-2	15
	Rank	5	3	2	1	1	1	5	3

Table 5.15: IGD Ranking for 15-objective DTLZ (continue)

Algorithm	Result	1	2	3	4	5	6	7	Overall
$KnMGPSO_R$	Wins	0	5	6	1	4	3	6	25
	Losses	4	0	1	2	0	0	0	7
	Difference	-4	5	5	-1	4	3	6	18
	Rank	5	2	2	7	1	1	1	2
$KnMGPSO_{RI}$	Wins	0	7	6	2	4	3	6	28
	Losses	4	0	1	1	0	0	0	6
	Difference	-4	7	5	1	4	3	6	22
	Rank	5	1	2	3	1	1	1	1
CDAS-SMPSO	Wins	7	3	0	0	1	0	0	11
	Losses	1	3	9	0	8	7	9	37
	Difference	6	0	-9	0	-7	-7	-9	-26
	Rank	2	7	10	4	9	10	10	9
KnEA	Wins	7	0	1	0	0	0	2	10
	Losses	1	9	8	4	9	6	4	41
	Difference	6	-9	-7	-4	-9	-6	-2	-31
	Rank	2	10	9	10	10	9	5	10
MOEA/DD	Wins	9	1	2	0	2	0	1	15
	Losses	0	8	7	0	7	0	8	30
	Difference	9	-7	-5	0	-5	0	-7	-15
	Rank	1	9	8	4	8	7	9	8
NSGA-III	Wins	6	2	9	0	3	1	2	23
	Losses	3	7	0	0	6	6	4	26
	Difference	3	-5	9	0	-3	-5	-2	-3
	Rank	4	8	1	4	7	8	5	7

Table 5.16: IGD Ranking for 3-objective WFG

Algorithm	**Result**	**3-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
$MGPSO_{STD}$	Wins	4	0	5	0	1	3	0	0	2	15
	Losses	4	5	1	7	0	5	4	4	4	34
	Difference	0	-5	4	-7	1	-2	-4	-4	-2	-19
	Rank	5	9	3	10	4	6	8	5	6	7
$MGPSO_{R}$	Wins	0	0	3	0	1	0	0	0	0	4
	Losses	8	4	3	6	1	8	7	4	4	45
	Difference	-8	-4	0	-6	0	-8	-7	-4	-4	-41
	Rank	9	7	6	8	5	9	9	5	7	9
$MGPSO_{RI}$	Wins	0	0	3	0	1	0	0	0	0	4
	Losses	8	6	3	6	1	8	7	4	6	49
	Difference	-8	-6	0	-6	0	-8	-7	-4	-6	-45
	Rank	9	10	6	8	5	9	9	5	9	10
$KnMGPSO_{STD}$	Wins	4	1	6	1	6	5	2	0	3	28
	Losses	4	4	1	4	0	4	4	4	4	29
	Difference	0	-3	5	-3	6	1	-2	-4	-1	-1
	Rank	5	6	2	7	1	5	5	5	5	5
$KnMGPSO_{R}$	Wins	2	0	3	3	2	2	2	0	0	14
	Losses	6	4	2	4	0	6	4	4	6	36
	Difference	-4	-4	1	-1	2	-4	-2	-4	-6	-22
	Rank	7	7	5	5	3	8	5	5	9	8
$KnMGPSO_{RI}$	Wins	2	2	3	3	4	2	2	0	0	18
	Losses	6	4	1	4	0	5	4	4	5	33
	Difference	-4	-2	2	-1	4	-3	-2	-4	-5	-15
	Rank	7	5	4	5	2	7	5	5	8	6

Table 5.16: IGD Ranking for 3-objective WFG (continue)

Algorithm	Result	3-objective WFG									Overall
		1	2	3	4	5	6	7	8	9	
CDAS-SMPSO	Wins	7	6	1	7	1	7	6	7	6	48
	Losses	0	2	7	1	5	2	2	0	3	22
	Difference	7	4	-6	6	-4	5	4	7	3	26
	Rank	1	4	8	2	9	3	3	1	4	3
KnEA	Wins	6	6	0	6	0	6	6	6	7	43
	Losses	3	1	8	2	9	3	2	3	2	33
	Difference	3	5	-8	4	-9	3	4	3	5	10
	Rank	4	3	10	4	10	4	3	4	3	4
MOEA/DD	Wins	7	7	0	6	2	8	8	7	9	54
	Losses	0	0	7	1	2	0	0	0	0	10
	Difference	7	7	-7	5	0	8	8	7	9	44
	Rank	1	2	9	3	5	1	1	1	1	2
NSGA-III	Wins	7	8	9	9	2	8	8	7	8	66
	Losses	0	0	0	0	2	0	0	0	1	3
	Difference	7	8	9	9	0	8	8	7	7	63
	Rank	1	1	1	1	5	1	1	1	2	1

Table 5.17: IGD Ranking for 5-objective WFG

Algorithm	Result	5-objective WFG									Overall
		1	2	3	4	5	6	7	8	9	
$MGPSO_{STD}$	Wins	0	0	2	0	4	2	0	0	0	8
	Losses	4	2	1	7	3	5	6	4	4	36
	Difference	-4	-2	1	-7	1	-3	-6	-4	-4	-28
	Rank	5	5	3	8	5	6	10	7	6	8

Table 5.17: IGD Ranking for 5-objective WFG (continue)

Algorithm	**Result**	**5-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
MGPSO_R	Wins	0	0	2	0	3	0	0	0	0	5
	Losses	4	3	2	7	3	8	5	6	5	43
	Difference	-4	-3	0	-7	0	-8	-5	-6	-5	-38
	Rank	5	8	5	8	6	9	8	10	10	10
MGPSO_{RI}	Wins	0	0	2	0	4	0	0	0	0	6
	Losses	4	3	2	7	2	8	5	4	4	39
	Difference	-4	-3	0	-7	2	-8	-5	-4	-4	-33
	Rank	5	8	5	8	4	9	8	7	6	9
KnMGPSO_{STD}	Wins	0	3	6	3	7	5	0	1	0	25
	Losses	4	2	1	4	0	4	4	4	4	27
	Difference	-4	1	5	-1	7	1	-4	-3	-4	-2
	Rank	5	3	2	5	1	5	7	5	6	5
KnMGPSO_R	Wins	0	0	2	3	6	2	3	1	1	18
	Losses	4	2	2	4	0	5	4	4	4	29
	Difference	-4	-2	0	-1	6	-3	-1	-3	-3	-11
	Rank	5	5	5	5	3	6	5	5	5	6
KnMGPSO_{RI}	Wins	0	1	2	3	7	2	1	0	0	16
	Losses	4	2	2	4	0	5	4	4	4	29
	Difference	-4	-1	0	-1	7	-3	-3	-4	-4	-13
	Rank	5	4	5	5	1	6	6	7	6	7
CDAS-SMPSO	Wins	8	9	2	8	2	8	8	6	7	58
	Losses	0	0	1	0	5	1	1	2	2	12
	Difference	8	9	1	8	-3	7	7	4	5	46
	Rank	1	1	3	1	7	2	2	3	3	2

Table 5.17: IGD Ranking for 5-objective WFG (continue)

Algorithm	Result	5-objective WFG 1	2	3	4	5	6	7	8	9	Overall
KnEA	Wins	6	0	0	6	0	6	6	6	6	36
	Losses	3	2	9	3	8	2	2	2	3	34
	Difference	3	-2	-9	3	-8	4	4	4	3	2
	Rank	4	5	10	4	9	3	3	3	4	4
MOEA/DD	Wins	7	0	1	7	0	6	6	8	8	43
	Losses	1	4	8	2	8	2	2	1	1	29
	Difference	6	-4	-7	5	-8	4	4	7	7	14
	Rank	3	10	9	3	9	3	3	2	2	3
NSGA-III	Wins	7	8	9	8	2	9	9	9	9	70
	Losses	0	1	0	0	6	0	0	0	0	7
	Difference	7	7	9	8	-4	9	9	9	9	63
	Rank	2	2	1	1	8	1	1	1	1	1

Table 5.18: IGD Ranking for 8-objective WFG

Algorithm	Result	8-objective WFG 1	2	3	4	5	6	7	8	9	Overall
MGPSO_{STD}	Wins	0	1	3	0	3	1	1	1	1	11
	Losses	4	5	2	7	1	3	3	5	3	33
	Difference	-4	-4	1	-7	2	-2	-2	-4	-2	-22
	Rank	8	7	5	8	4	6	4	8	7	8
MGPSO_{R}	Wins	0	1	3	0	3	1	1	1	1	11
	Losses	4	5	2	7	2	4	3	5	3	35
	Difference	-4	-4	1	-7	1	-3	-2	-4	-2	-24
	Rank	8	7	5	8	6	8	4	8	7	9

Table 5.18: IGD Ranking for 8-objective WFG (continue)

Algorithm	Result	8-objective WFG 1	2	3	4	5	6	7	8	9	Overall
$MGPSO_{RI}$	Wins	0	1	3	0	3	1	1	1	1	11
	Losses	4	5	2	7	1	4	3	4	2	32
	Difference	-4	-4	1	-7	2	-3	-2	-3	-1	-21
	Rank	8	7	5	8	4	8	4	7	5	7
$KnMGPSO_{STD}$	Wins	0	4	3	3	7	2	1	1	4	25
	Losses	3	1	1	4	0	3	3	3	1	19
	Difference	-3	3	2	-1	7	-1	-2	-2	3	6
	Rank	5	2	3	5	1	4	4	6	3	4
$KnMGPSO_{R}$	Wins	3	4	3	3	5	1	1	3	1	24
	Losses	3	1	1	4	0	3	3	3	2	20
	Difference	0	3	2	-1	5	-2	-2	0	-1	4
	Rank	4	2	3	5	2	6	4	5	5	6
$KnMGPSO_{RI}$	Wins	0	4	6	3	3	2	1	4	2	25
	Losses	3	1	1	4	0	3	3	3	1	19
	Difference	-3	3	5	-1	3	-1	-2	1	1	6
	Rank	5	2	2	5	3	4	4	4	4	4
CDAS-SMPSO	Wins	8	0	9	9	1	7	7	7	1	49
	Losses	0	9	0	0	7	1	1	0	4	22
	Difference	8	-9	9	9	-6	6	6	7	-3	27
	Rank	1	10	1	1	8	2	3	1	9	2
KnEA	Wins	7	4	0	6	0	0	0	0	0	17
	Losses	1	1	8	3	9	9	9	9	9	58
	Difference	6	3	-8	3	-9	-9	-9	-9	-9	-41
	Rank	3	2	10	4	10	10	10	10	10	10

Table 5.18: IGD Ranking for 8-objective WFG (continue)

Algorithm	Result	8-objective WFG 1	2	3	4	5	6	7	8	9	Overall
MOEA/DD	Wins	0	1	0	7	1	7	7	7	6	36
	Losses	3	1	7	1	7	1	0	0	0	20
	Difference	-3	0	-7	6	-6	6	7	7	6	16
	Rank	5	6	9	2	8	2	2	1	2	3
NSGA-III	Wins	7	9	1	7	3	9	8	7	8	59
	Losses	0	0	7	1	2	0	0	0	0	10
	Difference	7	9	-6	6	1	9	8	7	8	49
	Rank	2	1	8	2	6	1	1	1	1	1

Table 5.19: IGD Ranking for 10-objective WFG

Algorithm	Result	10-objective WFG 1	2	3	4	5	6	7	8	9	Overall
MGPSO$_{STD}$	Wins	0	2	3	0	4	2	0	0	4	15
	Losses	5	3	0	5	0	2	3	6	0	24
	Difference	-5	-1	3	-5	4	0	-3	-6	4	-9
	Rank	9	6	1	9	1	3	5	10	1	8
MGPSO$_{R}$	Wins	0	2	3	0	4	2	0	0	4	15
	Losses	3	3	0	6	0	2	3	4	0	21
	Difference	-3	-1	3	-6	4	0	-3	-4	4	-6
	Rank	6	6	1	10	1	3	5	9	1	7
MGPSO$_{RI}$	Wins	0	2	3	0	4	2	0	1	4	16
	Losses	3	3	0	4	0	2	3	2	0	17
	Difference	-3	-1	3	-4	4	0	-3	-1	4	-1
	Rank	6	6	1	7	1	3	5	5	1	6

Table 5.19: IGD Ranking for 10-objective WFG (continue)

Algorithm	Result	10-objective WFG									Overall
		1	2	3	4	5	6	7	8	9	
$KnMGPSO_{STD}$	Wins	0	5	3	0	4	2	0	1	4	19
	Losses	5	1	0	4	0	2	3	2	0	17
	Difference	-5	4	3	-4	4	0	-3	-1	4	2
	Rank	9	3	1	7	1	3	5	5	1	5
$KnMGPSO_{R}$	Wins	2	2	3	2	4	2	0	2	4	21
	Losses	3	2	0	4	0	2	3	2	0	16
	Difference	-1	0	3	-2	4	0	-3	0	4	5
	Rank	4	4	1	5	1	3	5	3	1	3
$KnMGPSO_{RI}$	Wins	2	2	3	1	4	2	0	2	4	20
	Losses	3	2	0	4	0	2	3	2	0	16
	Difference	-1	0	3	-3	4	0	-3	0	4	4
	Rank	4	4	1	6	1	3	5	3	1	4
CDAS-SMPSO	Wins	8	0	3	9	2	8	8	8	0	46
	Losses	0	8	0	0	6	0	0	0	6	20
	Difference	8	-8	3	9	-4	8	8	8	-6	26
	Rank	1	9	1	1	7	1	1	1	9	2
KnEA	Wins	7	7	0	6	1	0	0	0	0	21
	Losses	1	1	7	1	8	8	2	2	8	38
	Difference	6	6	-7	5	-7	-8	-2	-2	-8	-17
	Rank	3	2	9	2	9	9	4	7	10	9
MOEA/DD	Wins	0	0	0	6	0	0	6	0	1	13
	Losses	3	8	8	1	9	8	2	2	6	47
	Difference	-3	-8	-8	5	-9	-8	4	-2	-5	-34
	Rank	6	9	10	2	10	9	3	7	7	10

Table 5.19: IGD Ranking for 10-objective WFG (continue)

Algorithm	**Result**	**10-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
NSGA-III	Wins	7	9	1	6	2	8	8	8	1	50
	Losses	0	0	7	1	6	0	0	0	6	20
	Difference	7	9	-6	5	-4	8	8	8	-5	30
	Rank	2	1	8	2	7	1	1	1	7	1

Table 5.20: IGD Ranking for 15-objective WFG

Algorithm	**Result**	**15-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
MGPSO_{STD}	Wins	0	1	3	0	4	0	0	0	4	12
	Losses	1	1	0	4	0	2	4	4	0	16
	Difference	-1	0	3	-4	4	-2	-4	-4	4	-4
	Rank	5	3	1	6	1	3	5	5	1	5
MGPSO_{R}	Wins	0	1	3	0	4	0	0	0	4	12
	Losses	1	2	0	5	0	2	4	4	0	18
	Difference	-1	-1	3	-5	4	-2	-4	-4	4	-6
	Rank	5	9	1	10	1	3	5	5	1	9
MGPSO_{RI}	Wins	0	1	3	0	4	0	0	0	4	12
	Losses	1	1	0	4	0	2	4	4	0	16
	Difference	-1	0	3	-4	4	-2	-4	-4	4	-4
	Rank	5	3	1	6	1	3	5	5	1	5
KnMGPSO_{STD}	Wins	0	2	3	1	4	0	0	0	4	14
	Losses	1	1	0	4	0	2	4	4	0	16
	Difference	-1	1	3	-3	4	-2	-4	-4	4	-2
	Rank	5	2	1	5	1	3	5	5	1	4

Table 5.20: IGD Ranking for 15-objective WFG (continue)

Algorithm	Result	15-objective WFG 1	2	3	4	5	6	7	8	9	Overall
$KnMGPSO_R$	Wins	0	1	3	0	4	0	0	0	4	12
	Losses	1	1	0	4	0	2	4	4	0	16
	Difference	-1	0	3	-4	4	-2	-4	-4	4	-4
	Rank	5	3	1	6	1	3	5	5	1	5
$KnMGPSO_{RI}$	Wins	0	1	3	0	4	0	0	0	4	12
	Losses	1	1	0	4	0	2	4	4	0	16
	Difference	-1	0	3	-4	4	-2	-4	-4	4	-4
	Rank	5	3	1	6	1	3	5	5	1	5
CDAS-SMPSO	Wins	0	0	3	9	2	9	9	9	2	43
	Losses	0	9	0	0	6	0	0	0	6	21
	Difference	0	-9	3	9	-4	9	9	9	-4	22
	Rank	2	10	1	1	7	1	1	1	7	1
KnEA	Wins	6	1	0	6	1	0	6	6	2	28
	Losses	0	1	7	1	8	2	1	1	6	27
	Difference	6	0	-7	5	-7	-2	5	5	-4	1
	Rank	1	3	8	3	9	3	2	3	7	3
MOEA/DD	Wins	0	1	0	6	0	0	6	6	0	19
	Losses	0	1	7	2	9	2	1	2	8	32
	Difference	0	0	-7	4	-9	-2	5	4	-8	-13
	Rank	2	3	8	4	10	3	2	4	9	10
NSGA-III	Wins	0	9	0	7	2	8	6	7	0	39
	Losses	0	0	7	1	6	1	1	1	8	25
	Difference	0	9	-7	6	-4	7	5	6	-8	14
	Rank	2	1	8	2	7	2	2	2	9	2

5.4.1 Hypervolume Discussion

The overall HV rankings for each algorithm, shown in tables 5.1 to 5.10, are analyzed and discussed next.

The MGPSO$_{STD}$ ranked best overall three times with respect to HV. The MGPSO$_{STD}$ received the best overall rank for the 3-, 10-, and 15-objective DTLZ problems (tables 5.1, 5.4, and 5.5). The MGPSO$_{STD}$ ranked top-three overall six times in terms of HV (tables 5.1 to 5.5, and 5.10). The MGPSO$_{STD}$ ranked worst overall in terms of HV only for the 3-objective WFG problems (table 5.6). Therefore, the MGPSO$_{STD}$ scaled and performed very competitively with respect to HV, even compared to the MaOO benchmark algorithms.

The MGPSO$_R$ never performed best overall. The MGPSO$_R$ ranked third overall only twice in terms of HV, for the 8- and 10-objective WFG problems (tables 5.8 and 5.9). The worst overall rank received by the MGPSO$_R$ considering HV was seven, for the 5-objective DTLZ problems (table 5.2). The overall performance of the MGPSO$_R$ with respect to HV was medial. Therefore, the MGPSO$_R$ did not scale to many-objectives competitively with respect to HV.

The MGPSO$_{RI}$ also never received the best overall rank. The best overall rank received by the MGPSO$_{RI}$ in terms of HV was two, achieved for the 8-objective WFG problems (table 5.8). The MGPSO$_{RI}$ ranked top-three overall four times in terms of HV (tables 5.4, 5.5, 5.8, and 5.9). The worst overall rank obtained by the MGPSO$_{RI}$ with respect to HV was seven, obtained for the 5-objective DTLZ and WFG problems (tables 5.2 and 5.7). The overall performance of the MGPSO$_{RI}$ in terms of HV was also medial. Therefore, the MGPSO$_{RI}$ did not scale to many-objectives competitively with respect to HV. The MGPSO$_{RI}$ outperformed the MGPSO$_R$, ranking top-three overall twice as much in terms of HV.

The KnMGPSO$_{STD}$ ranked best overall twice in terms of HV, for the 8- and 10-objective WFG problems (tables 5.8 and 5.9). The KnMGPSO$_{STD}$ ranked top-three overall seven times with respect to HV; that is, the most compared to the other algorithms (tables 5.1, 5.2, 5.4, 5.5, and 5.8 to 5.10). The KnMGPSO$_{STD}$ obtained the worst overall rank twice with regards to HV, for the 3- and 5-objective WFG problems (tables 5.6 and 5.7). Therefore, the KnMGPSO$_{STD}$ scaled and performed very competitively in terms of HV.

The best and only top-three overall rank obtained by the KnMGPSO$_R$ with respect to HV was three, for the 10-objective WFG problems (table 5.9). The KnMGPSO$_R$ never ranked worst overall. The worst overall rank obtained by the KnMGPSO$_R$ in terms of HV was eight, for the 8-objective DTLZ problems (table 5.3). The KnMGPSO$_R$ performed towards the middle overall with respect to HV. Therefore, the KnMGPSO$_R$ did not scale to many-objectives competitively with respect to HV.

The KnMGPSO$_{RI}$ performed best overall once with respect to HV, for the 10-objective WFG problems (table 5.9). The KnMGPSO$_{RI}$ obtained top-three overall ranks with respect to HV twice (tables 5.6 and 5.9). The KnMGPSO$_{RI}$, like the KnMGPSO$_R$, also never ranked worst overall. The worst overall rank obtained by the KnMGPSO$_{RI}$ in terms of HV was eight, for the 5-objective DTLZ and WFG problems (tables 5.2 and 5.7). Therefore, the KnMGPSO$_{RI}$ did not perform competitively with regards to HV.

The best overall rank obtained by the CDAS-SMPSO algorithm with respect to HV was two, for the 3- and 5-objective WFG problems (tables 5.6 and 5.7). The CDAS-SMPSO algorithm ranked top-three overall twice in terms of HV (tables 5.6 and 5.7). The CDAS-SMPSO algorithm performed worst overall twice in terms of HV, for the 3- and 15-objective DTLZ problems (tables 5.1 and 5.5). The CDAS-SMPSO algorithm ranked towards the bottom overall in terms of HV often. Therefore, the CDAS-SMPSO algorithm did not scale well with regards to HV.

The KnEA ranked first overall twice in terms of HV, for the 3- and 5-objective WFG problems (tables 5.6 and 5.7). The KnEA ranked top-three overall three times with respect to HV (tables 5.1, 5.6 and 5.7). The KnEA performed worst overall more often than any other algorithm in terms of HV. That is, five times for the 5-, 8-, and 10-objective DTLZ problems and the 8- and 15-objective WFG problems (tables 5.2 to 5.4, 5.8, and 5.10). Therefore, the KnEA scaled and performed the worst in terms of HV. Also note that the KnMGPSO algorithm variants, which used suggested values for κ, outperformed the KnEA which used optimized (tuned) values for κ.

The MOEA/DD performed best overall twice in terms of HV, for the 5- and 8-objective DTLZ problems (tables 5.2 and 5.3). That is, the only two times that the MOEA/DD ranked top-three overall with respect to HV. The MOEA/DD obtained the worst overall rank once with respect to HV, for the 10-objective WFG problems (table 5.9). The MOEA/DD ranked near the bottom overall often with respect to HV. Therefore, the MOEA/DD did not perform or scale competitively with respect to HV.

The NSGA-III received the best overall rank in terms of HV once, for the 15-objective WFG problems (table 5.10). The NSGA-III ranked top-three overall four times with respect to HV (tables 5.2, 5.3, 5.7, and 5.10). The NSGA-III never performed worst overall. The worst overall rank obtained by the NSGA-III in terms of HV was nine, for the 3- and 15-objective DTLZ problems (tables 5.1 and 5.5). Therefore, the NSGA-III performed and scaled competitively in terms of HV, especially compared to the other MaOO algorithms.

5.4.2 Inverted Generational Distance Discussion

The overall IGD rankings for each algorithm, shown in tables 5.11 to 5.20, are analyzed and discussed next.

The best overall rank obtained by the MGPSO$_{STD}$ in terms of IGD was two, for the 10-objective DTLZ problems (table 5.14). That is, the only time the MGPSO$_{STD}$ ranked top-three overall in terms of IGD. The worst overall rank received by the MGPSO$_{STD}$ in terms of IGD was nine, for the 3-objective DTLZ problems (table 5.11). The MGPSO$_{STD}$ ranked in the bottom-half overall often in terms of IGD. Therefore, the MGPSO$_{STD}$ did not perform competitively with respect to IGD.

The MGPSO$_R$ never performed best overall. The best overall IGD rank obtained by the MGPSO$_R$ was five, for the 8-, 10-, and 15-objective DTLZ problems (tables 5.13 to 5.15). The MGPSO$_R$ performed worst overall only once in terms of IGD, for the 5-objective WFG problems (table 5.17). Therefore, the MGPSO$_R$ did not perform competitively with respect to IGD.

The MGPSO$_{RI}$ also never received the best overall rank. The best overall rank received by the MGPSO$_{RI}$ in terms of IGD was four, obtained for the 15-objective DTLZ problems (table 5.15). The MGPSO$_{RI}$ performed worst overall only once in terms of IGD for the 3-objective WFG problems (table 5.16). Therefore, the MGPSO$_{RI}$ did not perform competitively with respect to IGD. The overall performance of the MGPSO$_{STD}$, the MGPSO$_R$, and the MGPSO$_{RI}$ was comparable with respect to IGD.

The best overall rank obtained by the KnMGPSO$_{STD}$ with respect to IGD was two, for the 10-objective DTLZ problems (table 5.14). The KnMGPSO$_{STD}$ ranked top-three overall three times with respect to IGD (tables 5.13 to 5.15). The worst overall rank obtained by the KnMGPSO$_{STD}$ with respect to IGD was eight, for the 3-objective DTLZ problems (table 5.11). The KnMGPSO$_{STD}$, however, ranked in the top-half overall for all other problems with regards to IGD. Therefore, the KnMGPSO$_{STD}$ did not perform competitively although consistently.

The KnMGPSO$_R$ only performed best overall once with respect to IGD, that is, for the 10-objective DTLZ problems (table 5.14). The KnMGPSO$_R$ ranked top-three overall four times with respect to IGD (tables 5.18, 5.19, 5.14, and 5.15). The KnMGPSO$_R$ never ranked worst overall. The worst overall rank obtained by the KnMGPSO$_R$ in terms of IGD was eight, for the 3-objective WFG problems (table 5.16). Therefore, the KnMGPSO$_R$ scaled and performed somewhat competitively with respect to IGD.

The KnMGPSO$_{RI}$ received the best overall rank once with respect to IGD, for the 15-objective DTLZ problems (table 5.15). The KnMGPSO$_{RI}$ obtained top-three overall ranks with respect to IGD twice (tables 5.12, and 5.15). The KnMGPSO$_{RI}$, like the KnMGPSO$_R$, also never performed worst overall. The worst overall rank obtained by the KnMGPSO$_{RI}$ in terms of IGD was seven, for the 5-objective WFG problems (table 5.17). The KnMGPSO$_{RI}$ ranked near the middle overall often in terms of IGD. Therefore, the KnMGPSO$_{RI}$ did not perform competitively in terms of IGD.

The CDAS-SMPSO algorithm performed best overall only once in terms of IGD, for the 15-objective WFG problems (table 5.20). The CDAS-SMPSO algorithm ranked top-three overall five times in terms of IGD (tables 5.16 to 5.20). The CDAS-SMPSO algorithm performed worst overall most often with respect to IGD; that is, three times for the 3-, 5-, and 8-objective

DTLZ problems (tables 5.11 to 5.13). The CDAS-SMPSO algorithm ranked near the bottom overall in terms of IGD several times. The CDAS-SMPSO algorithm also ranked top-three overall often. Therefore, the CDAS-SMPSO algorithm performed inconsistently with regards to IGD.

The best overall rank obtained by the KnEA with respect to IGD was three, for the 3-objective DTLZ problems (table 5.11) and the 15-objective WFG problems (table 5.20). That is, the only two times the KnEA ranked top-three overall with respect to IGD. The KnEA performed worst overall most often (together with the CDAS-SMPSO algorithm) in terms of IGD. That is, three times for the 10- and 15-objective DTLZ problems and the 8-objective WFG problems (tables 5.14, 5.15, and 5.18). Therefore, the KnEA scaled and performed among the worst in terms of IGD. Also note that the KnMGPSO algorithm variants, which used suggested values for κ, outperformed the KnEA which used optimized (tuned) values for κ.

The MOEA/DD performed best overall once in terms of IGD, for the 3-objective DTLZ problems (table 5.11). The MOEA/DD ranked top-three overall six times with respect to IGD (tables 5.11 to 5.13, and 5.16 to 5.18). The MOEA/DD obtained the worst overall rank twice with respect to IGD, for the 10- and 15-objective WFG problems (tables 5.19 and 5.20). Therefore, the MOEA/DD performed and scaled somewhat competitively with respect to IGD.

The NSGA-III received the best overall rank with respect to IGD six times - more than any algorithm investigated in this chapter. That is, for the 5- and 8-objective DTLZ problems and the 3-, 5-, 8-, 10-, and 15-objective WFG problems (tables 5.12, 5.13, and 5.16 to 5.20). The NSGA-III also performed top-three overall the most in terms of IGD; that is, eight times (tables 5.11 to 5.13 and 5.16 to 5.20). The NSGA-III never performed worst overall. The worst overall rank obtained by the NSGA-III with regards to IGD was seven, for the 10- and 15-objective DTLZ problems (tables 4.14 and 4.15). Therefore, the NSGA-III was superior to all other algorithms investigated in this chapter in terms of IGD.

5.4.3 General Discussion

Some general findings with regards to tables 5.1 to 5.20 are discussed next.

The knee-points approach (the KnMGPSO algorithm) slightly improved the scalability of the MGPSO algorithm in terms of IGD, but not in terms of HV. The KnMGPSO$_{STD}$ was, however, very competitive in terms of HV. Also, note that the KnEA, which also uses knee-points, performed worst overall throughout this study (even with tuned control parameter values). Therefore, the results suggest that the knee-points approach is not as competitive compared to the mechanisms used by the other state-of-the-art MaOO algorithms.

The MGPSO$_{STD}$ scaled to many-objectives well in terms of HV, despite using the degrading Pareto-dominance relation. Recall that the Pareto-dominance relation struggles to distinguish solution desirability as the number of objectives for the problem continues to increase. The Pareto-dominance relation controls which solutions may be inserted into the archive but it is not used to select the actual archive guide. Also, the archive guide is not the only component in the MGPSO algorithm that guides the search. That is, the personal best and neighbourhood best positions also help guide the search. Therefore, the MGPSO algorithm is not affected that much by the many-objective weakness associated with the Pareto-dominance relation.

The use of different dynamic archive balance coefficient update strategies did not improve the performance of the MGPSO algorithm or the KnMGPSO algorithm significantly. The original STD approach was deemed the superior option in both the MGPSO algorithm and the KnMGPSO algorithm.

The results, discussed in the sections above, are summarized in tables 5.21 and 5.22. The tables contain the number of overall best, overall top-three, and overall worst ranks obtained by each algorithm with respect to either HV or IGD. Tables 5.23 and 5.24 contain the average overall rank obtained by each algorithm for each benchmark problem suite for each number of objectives with respect to either HV or IGD.

Table 5.21: HV Ranking Summary

Algorithm	**Number of overall best HV ranks**	**Number of overall HV ranks $\leq$ 3**	**Number of overall worst HV ranks**
MGPSO_{STD}	3	6	1
MGPSO_{R}	0	2	0
MGPSO_{RI}	0	4	0
KnMGPSO_{STD}	2	7	2
KnMGPSO_{R}	0	1	0
KnMGPSO_{RI}	1	2	0
CDAS-SMPSO	0	2	2
KnEA	2	3	5
MOEA/DD	2	2	1
NSGA-III	1	4	0

5.5 Summary

This chapter discussed the knee-points approach and introduced the knee-point driven multi-guide particle swarm optimization (KnMGPSO) algorithm. The KnMGPSO algorithm uses knee-points as a convergence metric alongside

Table 5.22: IGD Ranking Summary

Algorithm	**Number of overall best IGD ranks**	**Number of overall IGD ranks ≤ 3**	**Number of overall worst IGD ranks**
MGPSO_{STD}	0	1	0
MGPSO_{R}	0	0	1
MGPSO_{RI}	0	0	1
KnMGPSO_{STD}	0	3	0
KnMGPSO_{R}	1	4	0
KnMGPSO_{RI}	1	2	0
CDAS-SMPSO	1	5	3
KnEA	0	2	3
MOEA/DD	1	6	2
NSGA-III	6	8	0

the Pareto-dominance relation in order to increase the selection pressure towards the POF. This chapter empirically investigated the scalability of the multi-guide particle swarm optimization (MGPSO) [137] algorithm, the KnMGPSO algorithm, and benchmark algorithms on a set of benchmark problems by calculating, and statistically analyzing, the inverted generational distance (IGD) [25, 128] and hypervolume (HV) [181] performance measure values that were calculated on the normalized solutions without outliers. The benchmark algorithms included the controlling dominance area of solutions speed constraint multi-objective particle swarm optimization (CDAS-SMPSO) [33] algorithm, the knee-point driven evolutionary algorithm (KnEA) [174], the many-objective evolutionary algorithm based on dominance and decomposition (MOEA/DD) [95], and the reference-point based many-objective nondominated sorting genetic algorithm (NSGA-III) [38]. The benchmark problems with 3, 5, 8, 10, and 15 objectives were used to test the scalability of the algorithms. Three different archive balance coefficient update strategies were also investigated with the aim of improving scalability. These included the standard static archive balance coefficient update strategy (STD), the random dynamic archive balance coefficient update strategy (R), and the random per particle dynamic archive balance coefficient update strategy (RI) [53]. The results were presented in tables and discussed.

The MGPSO algorithm using the STD (MGPSO_{STD}) performed better with respect to HV than any other algorithm investigated in this chapter. The MGPSO algorithm using the R (MGPSO_{R}) and the MGPSO algorithm using the RI (MGPSO_{RI}) ranked in the middle overall in most cases. The KnMGPSO algorithm using the STD (KnMGPSO_{STD}) also performed competitively, considering the number of top-three overall HV rankings. The KnMGPSO algorithm using the R (KnMGPSO_{R}) and the KnMGPSO algorithm using the

Table 5.23: HV Ranking Averages

Algorithm	Benchmark Problems	n_m 3	5	8	10	15	Average
MGPSO_{STD}	DTLZ	1	2	2	1	1	1.4
	WFG	9	9	4	6	3	6.2
	Average Rank	5	5.5	3	3.5	2	3.8
MGPSO_{R}	DTLZ	4	7	5	4	4	4.8
	WFG	6	5	3	3	4	4.2
	Average Rank	5	6	4	3.5	4	4.5
MGPSO_{RI}	DTLZ	4	5	5	3	3	4
	WFG	5	7	2	3	5	4.4
	Average Rank	4.5	6	3.5	3	4	4.2
KnMGPSO_{STD}	DTLZ	3	2	4	2	2	2.6
	WFG	9	10	1	1	2	4.6
	Average Rank	6	6	2.5	1.5	2	3.6
KnMGPSO_{R}	DTLZ	7	5	8	6	4	6
	WFG	4	5	5	3	5	4.4
	Average Rank	5.5	5	6.5	4.5	4.5	5.2
KnMGPSO_{RI}	DTLZ	6	8	7	5	6	6.4
	WFG	3	8	6	1	7	5
	Average Rank	4.5	8	6.5	3	6.5	5.7
CDAS-SMPSO	DTLZ	10	9	9	8	10	9.2
	WFG	2	2	7	7	9	5.4
	Average Rank	6	5.5	8	7.5	9.5	7.3
KnEA	DTLZ	2	10	10	10	8	8
	WFG	1	1	10	8	10	6
	Average Rank	1.5	5.5	10	9	9	7
MOEA/DD	DTLZ	8	1	1	9	7	5.2
	WFG	8	4	9	10	8	7.8
	Average Rank	8	2.5	5	9.5	7.5	6.5
NSGA-III	DTLZ	9	2	3	7	9	6
	WFG	7	3	8	8	1	5.4
	Average Rank	8	2.5	5.5	7.5	5	5.7

Table 5.24: IGD Ranking Averages

Algorithm	**Benchmark Problems**	n_m **3**	**5**	**8**	**10**	**15**	**Average**
MGPSO_{STD}	DTLZ	9	8	8	2	6	6.6
	WFG	7	8	8	8	5	7.2
	Average Rank	8	8	8	5	5.5	6.9
MGPSO_{R}	DTLZ	7	6	5	5	5	5.6
	WFG	9	10	9	7	9	8.8
	Average Rank	8	8	7	6	7	7.2
MGPSO_{RI}	DTLZ	5	7	7	6	4	5.8
	WFG	10	9	7	6	5	7.4
	Average Rank	7.5	8	7	6	4.5	6.6
KnMGPSO_{STD}	DTLZ	8	5	3	2	3	4.2
	WFG	5	5	4	5	4	4.6
	Average Rank	6.5	5	3.5	3.5	3.5	4.4
KnMGPSO_{R}	DTLZ	4	4	3	1	2	2.8
	WFG	8	6	6	3	5	5.6
	Average Rank	6	5	4.5	2	3.5	4.2
KnMGPSO_{RI}	DTLZ	6	3	5	4	1	3.8
	WFG	6	7	4	4	5	5.2
	Average Rank	6	5	4.5	4	3	4.5
CDAS-SMPSO	DTLZ	10	10	10	9	9	9.6
	WFG	3	2	2	2	1	2
	Average Rank	6.5	6	6	5.5	5	5.8
KnEA	DTLZ	3	9	9	10	10	8.2
	WFG	4	4	10	9	3	6
	Average Rank	3.5	6.5	9.5	9.5	6.5	7.1
MOEA/DD	DTLZ	1	2	2	8	8	4.2
	WFG	2	3	3	10	10	5.6
	Average Rank	1.5	2.5	2.5	9	9	4.9
NSGA-III	DTLZ	2	1	1	7	7	3.6
	WFG	1	1	1	1	2	1.2
	Average Rank	1.5	1	1	4	4.5	2.4

RI (KnMGPSO$_{RI}$) performed somewhat competitively in terms of IGD and never ranked worst overall for both HV and IGD. The dynamic archive balance coefficient update strategies did not improve the scalability of MGPSO algorithm or the KnMGPSO algorithm. However, the KnMGPSO$_R$ and the KnMGPSO$_{RI}$ did outperform the MGPSO$_R$ and the MGPSO$_{RI}$ in terms of IGD.

The knee-points approach used by the KnMGPSO algorithm did not significantly improve algorithm performance. The knee-points approach was only able to barely improve the scalability of the MGPSO algorithm in terms of IGD.

In terms of the benchmark algorithms, the CDAS-SMPSO algorithm did not perform competitively, as it ranked inconsistently overall. The KnEA performed worst overall, ranking last overall most-often compared to the other algorithms. The MOEA/DD performed somewhat competitively in terms of IGD but not in terms of HV. The NSGA-III outperformed all the investigated algorithms in terms of IGD, ranking best overall and top-three overall more than any other algorithm. The NSGA-III also never ranked worst overall.

Chapter 6

Partial-dominance versus Knee-points

"The chief cause for failure and unhappiness is trading what you want most for what you want right now."

— Zig Ziglar

This chapter compares the scalability of the PMGPSO algorithm and the KnMGPSO algorithm. That is, this chapter investigates whether either of the MGPSO algorithm adaptations is best. The empirical process followed is presented in section 6.1. Section 6.2 presents and discusses the results. Finally, section 6.3 summarizes this chapter.

6.1 Empirical Process

Note that this chapter uses the same empirical proces as in sections 4.3 and 5.3. The algorithms and parameter tuning approach is discussed in section 6.1.1.

6.1.1 Algorithms and Parameter Tuning Approach

For this part of the study, the following algorithms were investigated:

1. PMGPSO$_{STD}$
2. PMGPSO$_{R}$
3. PMGPSO$_{RI}$
4. KnMGPSO$_{STD}$
5. KnMGPSO$_{R}$
6. KnMGPSO$_{RI}$

Note that this chapter uses the same parameter settings for the PMGPSO and KnMGPSO algorithms as in chapters 4 and 5.

6.2 Results and Discussion

The results are shown in tables 6.1 to 6.20 containing the overall wins, losses, difference, and rank across the problems for each algorithm. Note that the top-three best overall ranks are highlighted for each table. The HV and IGD performance measure values can be viewed in Appendix F. Sections 6.2.1 and 6.2.2 discuss the findings with respect to HV and IGD respectively. Finally, section 6.2.3 provides some general remarks and summarizes the findings.

Table 6.1: HV Ranking for 3-objective DTLZ

Algorithm	**Result**	**3-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$PMGPSO_{STD}$	Wins	2	2	2	0	2	0	0	8
	Losses	3	3	3	0	3	3	3	18
	Difference	-1	-1	-1	0	-1	-3	-3	-10
	Rank	4	4	4	3	4	4	4	4
$PMGPSO_{R}$	Wins	0	0	0	1	0	0	0	1
	Losses	4	4	4	0	4	3	3	22
	Difference	-4	-4	-4	1	-4	-3	-3	-21
	Rank	5	5	5	1	5	4	4	5
$PMGPSO_{RI}$	Wins	0	0	0	1	0	0	0	1
	Losses	4	4	4	0	4	3	3	22
	Difference	-4	-4	-4	1	-4	-3	-3	-21
	Rank	5	5	5	1	5	4	4	5
$KnMGPSO_{STD}$	Wins	5	3	5	0	3	3	3	22
	Losses	0	0	0	0	2	2	0	4
	Difference	5	3	5	0	1	1	3	18
	Rank	1	1	1	3	3	3	1	1
$KnMGPSO_{R}$	Wins	3	3	3	0	4	4	3	20
	Losses	1	0	1	0	0	0	0	2
	Difference	2	3	2	0	4	4	3	18
	Rank	2	1	2	3	1	1	1	1

Table 6.1: HV Ranking for 3-objective DTLZ (continue)

Algorithm	**Result**	**3-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$KnMGPSO_{RI}$	Wins	3	3	3	0	4	4	3	20
	Losses	1	0	1	2	0	0	0	4
	Difference	2	3	2	-2	4	4	3	16
	Rank	2	1	2	6	1	1	1	3

Table 6.2: HV Ranking for 5-objective DTLZ

Algorithm	**Result**	**5-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$PMGPSO_{STD}$	Wins	0	0	2	2	0	0	0	4
	Losses	3	3	3	3	3	3	3	21
	Difference	-3	-3	-1	-1	-3	-3	-3	-17
	Rank	4	4	4	4	4	4	4	4
$PMGPSO_{R}$	Wins	0	0	0	0	0	0	0	0
	Losses	3	3	4	4	3	3	3	23
	Difference	-3	-3	-4	-4	-3	-3	-3	-23
	Rank	4	4	5	5	4	4	4	5
$PMGPSO_{RI}$	Wins	0	0	0	0	0	0	0	0
	Losses	3	3	4	4	3	3	3	23
	Difference	-3	-3	-4	-4	-3	-3	-3	-23
	Rank	4	4	5	5	4	4	4	5
$KnMGPSO_{STD}$	Wins	5	3	5	3	3	3	4	26
	Losses	0	0	0	0	0	0	0	0
	Difference	5	3	5	3	3	3	4	26
	Rank	1	1	1	1	1	1	1	1

Table 6.2: HV Ranking for 5-objective DTLZ (continue)

Algorithm	**Result**	**5-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$KnMGPSO_R$	Wins	3	3	3	3	3	3	3	21
	Losses	1	0	1	0	0	0	1	3
	Difference	2	3	2	3	3	3	2	18
	Rank	2	1	2	1	1	1	3	3
$KnMGPSO_{RI}$	Wins	3	3	3	3	3	3	3	21
	Losses	1	0	1	0	0	0	0	2
	Difference	2	3	2	3	3	3	3	19
	Rank	2	1	2	1	1	1	2	2

Table 6.3: HV Ranking for 8-objective DTLZ

Algorithm	**Result**	**8-objective DTLZ**							**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	
$PMGPSO_{STD}$	Wins	0	0	0	2	2	2	0	6
	Losses	3	3	3	1	3	0	0	13
	Difference	-3	-3	-3	1	-1	2	0	-7
	Rank	4	4	4	2	4	1	2	4
$PMGPSO_R$	Wins	0	0	0	0	0	0	0	0
	Losses	3	3	3	4	4	0	1	18
	Difference	-3	-3	-3	-4	-4	0	-1	-18
	Rank	4	4	4	5	5	2	3	5
$PMGPSO_{RI}$	Wins	0	0	0	0	0	0	0	0
	Losses	3	3	3	4	4	0	1	18
	Difference	-3	-3	-3	-4	-4	0	-1	-18
	Rank	4	4	4	5	5	2	3	5

Table 6.3: HV Ranking for 8-objective DTLZ (continue)

Algorithm	Result	8-objective DTLZ 1	2	3	4	5	6	7	Overall
$KnMGPSO_{STD}$	Wins	4	3	3	5	5	0	4	24
	Losses	0	0	0	0	0	0	0	0
	Difference	4	3	3	5	5	0	4	24
	Rank	1	1	1	1	1	2	1	1
$KnMGPSO_{R}$	Wins	3	3	3	2	3	0	0	14
	Losses	0	0	0	1	1	1	1	4
	Difference	3	3	3	1	2	-1	-1	10
	Rank	2	1	1	2	2	5	3	2
$KnMGPSO_{RI}$	Wins	3	3	3	2	3	0	0	14
	Losses	1	0	0	1	1	1	1	5
	Difference	2	3	3	1	2	-1	-1	9
	Rank	3	1	1	2	2	5	3	3

Table 6.4: HV Ranking for 10-objective DTLZ

Algorithm	Result	10-objective DTLZ 1	2	3	4	5	6	7	Overall
$PMGPSO_{STD}$	Wins	0	0	0	2	0	1	4	7
	Losses	3	3	3	1	3	0	0	13
	Difference	-3	-3	-3	1	-3	1	4	-6
	Rank	4	4	4	2	4	2	1	4
$PMGPSO_{R}$	Wins	0	0	0	0	0	2	0	2
	Losses	3	3	3	3	3	0	2	17
	Difference	-3	-3	-3	-3	-3	2	-2	-15
	Rank	4	4	4	6	4	1	3	5

Table 6.4: HV Ranking for 10-objective DTLZ (continue)

Algorithm	Result	10-objective DTLZ 1	2	3	4	5	6	7	Overall
$PMGPSO_{RI}$	Wins	0	0	0	0	0	0	0	0
	Losses	3	3	3	2	3	0	2	16
	Difference	-3	-3	-3	-2	-3	0	-2	-16
	Rank	4	4	4	5	4	3	3	6
$KnMGPSO_{STD}$	Wins	3	3	3	5	4	0	4	22
	Losses	0	0	0	0	0	0	0	0
	Difference	3	3	3	5	4	0	4	22
	Rank	1	1	1	1	1	3	1	1
$KnMGPSO_{R}$	Wins	3	3	3	0	3	0	0	12
	Losses	0	0	0	1	0	1	2	4
	Difference	3	3	3	-1	3	-1	-2	8
	Rank	1	1	1	4	2	5	3	2
$KnMGPSO_{RI}$	Wins	3	3	3	1	3	0	0	13
	Losses	0	0	0	1	1	2	2	6
	Difference	3	3	3	0	2	-2	-2	7
	Rank	1	1	1	3	3	6	3	3

Table 6.5: HV Ranking for 15-objective DTLZ

Algorithm	Result	15-objective DTLZ 1	2	3	4	5	6	7	Overall
$PMGPSO_{STD}$	Wins	0	0	0	4	0	2	4	10
	Losses	3	3	3	1	3	0	0	13
	Difference	-3	-3	-3	3	-3	2	4	-3
	Rank	4	4	4	2	4	1	1	4

Table 6.5: HV Ranking for 15-objective DTLZ (continue)

Algorithm	Result	15-objective DTLZ 1	2	3	4	5	6	7	Overall
PMGPSO$_R$	Wins	0	0	0	0	0	2	0	2
	Losses	3	3	3	2	3	0	2	16
	Difference	-3	-3	-3	-2	-3	2	-2	-14
	Rank	4	4	4	5	4	1	3	5
PMGPSO$_{RI}$	Wins	0	0	0	0	0	0	0	0
	Losses	3	3	3	4	3	0	2	18
	Difference	-3	-3	-3	-4	-3	0	-2	-18
	Rank	4	4	4	6	4	3	3	6
KnMGPSO$_{STD}$	Wins	3	3	3	5	3	0	4	21
	Losses	0	0	0	0	0	0	0	0
	Difference	3	3	3	5	3	0	4	21
	Rank	1	1	1	1	1	3	1	1
KnMGPSO$_R$	Wins	3	3	3	1	3	0	0	13
	Losses	0	0	0	2	0	2	2	6
	Difference	3	3	3	-1	3	-2	-2	7
	Rank	1	1	1	3	1	5	3	2
KnMGPSO$_{RI}$	Wins	3	3	3	1	3	0	0	13
	Losses	0	0	0	2	0	2	2	6
	Difference	3	3	3	-1	3	-2	-2	7
	Rank	1	1	1	3	1	5	3	2

Table 6.6: HV Ranking for 3-objective WFG

Algorithm	Result	3-objective WFG 1	2	3	4	5	6	7	8	9	Overall
$PMGPSO_{STD}$	Wins	0	0	0	0	0	0	2	0	0	2
	Losses	3	3	3	3	3	3	0	3	3	24
	Difference	-3	-3	-3	-3	-3	-3	2	-3	-3	-22
	Rank	4	4	4	4	4	4	1	4	4	4
$PMGPSO_{R}$	Wins	0	0	0	0	0	0	0	0	0	0
	Losses	3	3	3	3	3	3	4	3	3	28
	Difference	-3	-3	-3	-3	-3	-3	-4	-3	-3	-28
	Rank	4	4	4	4	4	4	5	4	4	5
$PMGPSO_{RI}$	Wins	0	0	0	0	0	0	0	0	0	0
	Losses	3	3	3	3	3	3	4	3	3	28
	Difference	-3	-3	-3	-3	-3	-3	-4	-3	-3	-28
	Rank	4	4	4	4	4	4	5	4	4	5
$KnMGPSO_{STD}$	Wins	3	3	3	3	3	3	2	3	3	26
	Losses	2	1	2	0	0	2	0	0	2	9
	Difference	1	2	1	3	3	1	2	3	1	17
	Rank	3	3	3	1	1	3	1	1	3	3
$KnMGPSO_{R}$	Wins	4	3	4	3	3	4	2	3	4	30
	Losses	0	0	0	0	0	0	0	0	0	0
	Difference	4	3	4	3	3	4	2	3	4	30
	Rank	1	2	1	1	1	1	1	1	1	2
$KnMGPSO_{RI}$	Wins	4	4	4	3	3	4	2	3	4	31
	Losses	0	0	0	0	0	0	0	0	0	0
	Difference	4	4	4	3	3	4	2	3	4	31
	Rank	1	1	1	1	1	1	1	1	1	1

Table 6.7: HV Ranking for 5-objective WFG

Algorithm	**Result**	**5-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
PMGPSO$_{STD}$	Wins	0	2	2	0	0	0	0	0	0	4
	Losses	3	3	3	3	0	3	0	3	3	21
	Difference	-3	-1	-1	-3	0	-3	0	-3	-3	-17
	Rank	4	4	4	4	1	4	4	4	4	4
PMGPSO$_{R}$	Wins	0	0	0	0	0	0	0	0	0	0
	Losses	3	4	4	3	0	3	3	3	3	26
	Difference	-3	-4	-4	-3	0	-3	-3	-3	-3	-26
	Rank	4	5	5	4	1	4	5	4	4	5
PMGPSO$_{RI}$	Wins	0	0	0	0	0	0	0	0	0	0
	Losses	3	4	4	3	0	3	3	3	3	26
	Difference	-3	-4	-4	-3	0	-3	-3	-3	-3	-26
	Rank	4	5	5	4	1	4	5	4	4	5
KnMGPSO$_{STD}$	Wins	3	3	3	3	0	3	2	3	3	23
	Losses	2	2	2	0	0	2	0	0	0	8
	Difference	1	1	1	3	0	1	2	3	3	15
	Rank	3	3	3	1	1	3	1	1	1	3
KnMGPSO$_{R}$	Wins	4	4	4	3	0	4	2	3	3	27
	Losses	0	0	0	0	0	0	0	0	0	0
	Difference	4	4	4	3	0	4	2	3	3	27
	Rank	1	1	1	1	1	1	1	1	1	1
KnMGPSO$_{RI}$	Wins	4	4	4	3	0	4	2	3	3	27
	Losses	0	0	0	0	0	0	0	0	0	0
	Difference	4	4	4	3	0	4	2	3	3	27
	Rank	1	1	1	1	1	1	1	1	1	1

Table 6.8: HV Ranking for 8-objective WFG

Algorithm	**Result**	**8-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
PMGPSO_{STD}	Wins	0	2	1	0	4	0	0	0	0	7
	Losses	3	3	3	0	0	3	1	1	3	17
	Difference	-3	-1	-2	0	4	-3	-1	-1	-3	-10
	Rank	4	4	4	2	1	4	4	4	4	4
PMGPSO_{R}	Wins	0	0	0	0	2	0	0	0	0	2
	Losses	3	4	4	1	1	3	2	3	3	24
	Difference	-3	-4	-4	-1	1	-3	-2	-3	-3	-22
	Rank	4	5	6	5	3	4	6	5	4	6
PMGPSO_{RI}	Wins	0	0	0	0	2	0	0	0	0	2
	Losses	3	4	3	1	1	3	1	3	3	22
	Difference	-3	-4	-3	-1	1	-3	-1	-3	-3	-20
	Rank	4	5	5	5	3	4	4	5	4	5
KnMGPSO_{STD}	Wins	3	3	5	2	2	3	3	5	3	29
	Losses	2	1	0	0	0	0	0	0	0	3
	Difference	1	2	5	2	2	3	3	5	3	26
	Rank	3	3	1	1	2	1	1	1	1	1
KnMGPSO_{R}	Wins	4	3	3	0	0	3	1	2	3	19
	Losses	0	0	1	0	4	0	0	1	0	6
	Difference	4	3	2	0	-4	3	1	1	3	13
	Rank	1	2	2	2	5	1	2	2	1	2
KnMGPSO_{RI}	Wins	4	4	3	0	0	3	0	2	3	19
	Losses	0	0	1	0	4	0	0	1	0	6
	Difference	4	4	2	0	-4	3	0	1	3	13
	Rank	1	1	2	2	5	1	3	2	1	2

Table 6.9: HV Ranking for 10-objective WFG

Algorithm	**Result**	**10-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
PMGPSO$_{STD}$	Wins	0	0	2	0	4	0	0	2	0	8
	Losses	5	3	3	0	0	1	1	0	2	15
	Difference	-5	-3	-1	0	4	-1	-1	2	-2	-7
	Rank	6	4	4	1	1	4	3	2	4	4
PMGPSO$_{R}$	Wins	1	0	0	0	1	0	0	0	0	2
	Losses	3	3	4	0	2	2	2	4	2	22
	Difference	-2	-3	-4	0	-1	-2	-2	-4	-2	-20
	Rank	4	4	5	1	4	5	6	5	4	6
PMGPSO$_{RI}$	Wins	1	0	0	0	2	0	0	0	0	3
	Losses	3	3	4	0	2	3	1	4	2	22
	Difference	-2	-3	-4	0	0	-3	-1	-4	-2	-19
	Rank	4	4	5	1	3	6	3	5	4	5
KnMGPSO$_{STD}$	Wins	3	3	5	0	4	1	4	3	0	23
	Losses	2	0	0	0	0	0	0	0	0	2
	Difference	1	3	5	0	4	1	4	3	0	21
	Rank	3	1	1	1	1	3	1	1	3	1
KnMGPSO$_{R}$	Wins	4	3	3	0	0	2	1	2	3	18
	Losses	0	0	1	0	3	0	0	1	0	5
	Difference	4	3	2	0	-3	2	1	1	3	13
	Rank	1	1	2	1	5	2	2	4	1	2
KnMGPSO$_{RI}$	Wins	4	3	3	0	0	3	0	2	3	18
	Losses	0	0	1	0	4	0	1	0	0	6
	Difference	4	3	2	0	-4	3	-1	2	3	12
	Rank	1	1	2	1	6	1	3	2	1	3

Table 6.10: HV Ranking for 15-objective WFG

Algorithm	Result	15-objective WFG 1	2	3	4	5	6	7	8	9	Overall
$PMGPSO_{STD}$	Wins	0	0	0	0	4	0	0	0	0	4
	Losses	3	3	3	3	0	3	0	0	0	15
	Difference	-3	-3	-3	-3	4	-3	0	0	0	-11
	Rank	4	4	4	4	1	5	2	4	1	4
$PMGPSO_{R}$	Wins	0	0	0	0	2	0	0	0	0	2
	Losses	3	3	3	3	1	3	1	3	0	20
	Difference	-3	-3	-3	-3	1	-3	-1	-3	0	-18
	Rank	4	4	4	4	3	5	5	6	1	6
$PMGPSO_{RI}$	Wins	0	0	0	0	1	0	0	0	0	1
	Losses	3	3	3	3	1	1	1	2	0	17
	Difference	-3	-3	-3	-3	0	-1	-1	-2	0	-16
	Rank	4	4	4	4	4	4	5	5	1	5
$KnMGPSO_{STD}$	Wins	3	3	3	3	2	2	0	2	0	18
	Losses	2	0	0	0	0	0	0	0	0	2
	Difference	1	3	3	3	2	2	0	2	0	16
	Rank	3	1	1	1	2	2	2	1	1	1
$KnMGPSO_{R}$	Wins	4	3	3	3	0	3	0	2	0	18
	Losses	0	0	0	0	3	0	0	0	0	3
	Difference	4	3	3	3	-3	3	0	2	0	15
	Rank	1	1	1	1	5	1	2	1	1	2
$KnMGPSO_{RI}$	Wins	4	3	3	3	0	2	2	1	0	18
	Losses	0	0	0	0	4	0	0	0	0	4
	Difference	4	3	3	3	-4	2	2	1	0	14
	Rank	1	1	1	1	6	2	1	3	1	3

Table 6.11: IGD Ranking for 3-objective DTLZ

Algorithm	**Result**	**3-objective DTLZ 1**	**2**	**3**	**4**	**5**	**6**	**7**	**Overall**
PMGPSO_{STD}	Wins	0	3	3	3	3	3	3	18
	Losses	2	0	0	1	0	0	0	3
	Difference	-2	3	3	2	3	3	3	15
	Rank	3	1	1	3	1	1	1	3
PMGPSO_{R}	Wins	0	3	3	4	3	3	3	19
	Losses	2	0	0	0	0	0	0	2
	Difference	-2	3	3	4	3	3	3	17
	Rank	3	1	1	1	1	1	1	1
PMGPSO_{RI}	Wins	0	3	3	3	3	3	3	18
	Losses	2	0	0	0	0	0	0	2
	Difference	-2	3	3	3	3	3	3	16
	Rank	3	1	1	2	1	1	1	2
KnMGPSO_{STD}	Wins	0	0	0	0	1	2	1	4
	Losses	2	3	5	3	3	3	3	22
	Difference	-2	-3	-5	-3	-2	-1	-2	-18
	Rank	3	4	6	4	4	4	4	6
KnMGPSO_{R}	Wins	4	0	1	0	0	0	0	5
	Losses	0	3	3	3	3	4	4	20
	Difference	4	-3	-2	-3	-3	-4	-4	-15
	Rank	1	4	4	4	5	5	6	4
KnMGPSO_{RI}	Wins	4	0	1	0	0	0	0	5
	Losses	0	3	3	3	4	4	3	20
	Difference	4	-3	-2	-3	-4	-4	-3	-15
	Rank	1	4	4	4	6	5	5	4

Table 6.12: IGD Ranking for 5-objective DTLZ

Algorithm	**Result**	**5-objective DTLZ 1**	**2**	**3**	**4**	**5**	**6**	**7**	**Overall**
$PMGPSO_{STD}$	Wins	0	3	3	0	3	0	3	12
	Losses	3	0	0	0	0	0	1	4
	Difference	-3	3	3	0	3	0	2	8
	Rank	4	1	1	1	1	1	3	3
$PMGPSO_{R}$	Wins	0	3	3	0	3	0	3	12
	Losses	3	0	0	0	0	0	0	3
	Difference	-3	3	3	0	3	0	3	9
	Rank	4	1	1	1	1	1	2	2
$PMGPSO_{RI}$	Wins	0	3	3	0	3	0	4	13
	Losses	3	0	0	0	0	0	0	3
	Difference	-3	3	3	0	3	0	4	10
	Rank	4	1	1	1	1	1	1	1
$KnMGPSO_{STD}$	Wins	3	0	0	0	0	0	0	3
	Losses	0	3	3	0	3	0	5	14
	Difference	3	-3	-3	0	-3	0	-5	-11
	Rank	1	4	4	1	4	1	6	6
$KnMGPSO_{R}$	Wins	3	0	0	0	0	0	1	4
	Losses	0	3	3	0	3	0	3	12
	Difference	3	-3	-3	0	-3	0	-2	-8
	Rank	1	4	4	1	4	1	4	4
$KnMGPSO_{RI}$	Wins	3	0	0	0	0	0	1	4
	Losses	0	3	3	0	3	0	3	12
	Difference	3	-3	-3	0	-3	0	-2	-8
	Rank	1	4	4	1	4	1	4	4

Table 6.13: IGD Ranking for 8-objective DTLZ

Algorithm	**Result**	**8-objective DTLZ 1**	**2**	**3**	**4**	**5**	**6**	**7**	**Overall**
$PMGPSO_{STD}$	Wins	0	3	3	0	3	0	3	12
	Losses	3	0	0	0	0	0	0	3
	Difference	-3	3	3	0	3	0	3	9
	Rank	4	1	1	2	1	1	1	1
$PMGPSO_{R}$	Wins	0	3	3	0	3	0	3	12
	Losses	3	0	0	0	0	0	0	3
	Difference	-3	3	3	0	3	0	3	9
	Rank	4	1	1	2	1	1	1	1
$PMGPSO_{RI}$	Wins	0	3	3	0	3	0	3	12
	Losses	3	0	0	1	0	0	0	4
	Difference	-3	3	3	-1	3	0	3	8
	Rank	4	1	1	4	1	1	1	3
$KnMGPSO_{STD}$	Wins	3	0	0	3	0	0	0	6
	Losses	0	3	3	0	3	0	5	14
	Difference	3	-3	-3	3	-3	0	-5	-8
	Rank	1	4	4	1	4	1	6	4
$KnMGPSO_{R}$	Wins	3	0	0	0	0	0	1	4
	Losses	0	3	3	1	3	0	3	13
	Difference	3	-3	-3	-1	-3	0	-2	-9
	Rank	1	4	4	4	4	1	4	5
$KnMGPSO_{RI}$	Wins	3	0	0	0	0	0	1	4
	Losses	0	3	3	1	3	0	3	13
	Difference	3	-3	-3	-1	-3	0	-2	-9
	Rank	1	4	4	4	4	1	4	5

Table 6.14: IGD Ranking for 10-objective DTLZ

Algorithm	**Result**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Overall**
		10-objective DTLZ							
$PMGPSO_{STD}$	Wins	0	0	3	3	3	0	3	12
	Losses	3	0	0	0	0	0	0	3
	Difference	-3	0	3	3	3	0	3	9
	Rank	4	3	1	1	1	1	1	3
$PMGPSO_{R}$	Wins	0	3	3	3	3	0	3	15
	Losses	3	0	0	0	0	0	0	3
	Difference	-3	3	3	3	3	0	3	12
	Rank	4	1	1	1	1	1	1	1
$PMGPSO_{RI}$	Wins	0	2	3	3	3	0	3	14
	Losses	3	0	0	0	0	0	0	3
	Difference	-3	2	3	3	3	0	3	11
	Rank	4	2	1	1	1	1	1	2
$KnMGPSO_{STD}$	Wins	3	0	0	0	0	0	0	3
	Losses	0	2	3	3	3	0	5	16
	Difference	3	-2	-3	-3	-3	0	-5	-13
	Rank	1	5	4	4	4	1	6	6
$KnMGPSO_{R}$	Wins	3	0	0	0	0	0	1	4
	Losses	0	2	3	3	3	0	3	14
	Difference	3	-2	-3	-3	-3	0	-2	-10
	Rank	1	5	4	4	4	1	4	5
$KnMGPSO_{RI}$	Wins	3	0	0	0	0	0	1	4
	Losses	0	1	3	3	3	0	3	13
	Difference	3	-1	-3	-3	-3	0	-2	-9
	Rank	1	4	4	4	4	1	4	4

Table 6.15: IGD Ranking for 15-objective DTLZ

Algorithm	**Result**	**15-objective DTLZ 1**	**2**	**3**	**4**	**5**	**6**	**7**	**Overall**
$PMGPSO_{STD}$	Wins	0	0	0	2	3	0	3	8
	Losses	3	3	0	0	0	0	0	6
	Difference	-3	-3	0	2	3	0	3	2
	Rank	4	4	1	3	1	3	1	3
$PMGPSO_{R}$	Wins	0	0	0	3	3	1	3	10
	Losses	3	3	0	0	0	0	0	6
	Difference	-3	-3	0	3	3	1	3	4
	Rank	4	4	1	1	1	1	1	1
$PMGPSO_{RI}$	Wins	0	0	0	3	3	1	3	10
	Losses	3	3	0	0	0	0	0	6
	Difference	-3	-3	0	3	3	1	3	4
	Rank	4	4	1	1	1	1	1	1
$KnMGPSO_{STD}$	Wins	3	3	0	0	0	0	0	6
	Losses	0	0	0	3	3	0	5	11
	Difference	3	3	0	-3	-3	0	-5	-5
	Rank	1	1	1	5	4	3	6	6
$KnMGPSO_{R}$	Wins	3	3	0	0	0	0	1	7
	Losses	0	0	0	2	3	2	3	10
	Difference	3	3	0	-2	-3	-2	-2	-3
	Rank	1	1	1	4	4	6	4	5
$KnMGPSO_{RI}$	Wins	3	3	0	0	0	0	1	7
	Losses	0	0	0	3	3	0	3	9
	Difference	3	3	0	-3	-3	0	-2	-2
	Rank	1	1	1	5	4	3	4	4

Table 6.16: IGD Ranking for 3-objective WFG

Algorithm	Result	3-objective WFG 1	2	3	4	5	6	7	8	9	Overall
PMGPSO_{STD}	Wins	3	3	3	3	5	3	3	3	4	30
	Losses	0	0	0	0	0	0	2	0	0	2
	Difference	3	3	3	3	5	3	1	3	4	28
	Rank	1	1	1	1	1	1	3	1	1	1
PMGPSO_{R}	Wins	3	3	3	3	3	3	4	3	3	28
	Losses	0	0	0	0	1	0	0	0	2	3
	Difference	3	3	3	3	2	3	4	3	1	25
	Rank	1	1	1	1	2	1	1	1	3	3
PMGPSO_{RI}	Wins	3	3	3	3	3	3	4	3	4	29
	Losses	0	0	0	0	1	0	0	0	0	1
	Difference	3	3	3	3	2	3	4	3	4	28
	Rank	1	1	1	1	2	1	1	1	1	1
KnMGPSO_{STD}	Wins	2	0	2	0	1	2	0	0	2	9
	Losses	3	3	3	3	3	3	3	3	3	27
	Difference	-1	-3	-1	-3	-2	-1	-3	-3	-1	-18
	Rank	4	4	4	4	4	4	4	4	4	4
KnMGPSO_{R}	Wins	0	0	0	0	0	0	0	0	0	0
	Losses	4	3	4	3	4	4	3	3	4	32
	Difference	-4	-3	-4	-3	-4	-4	-3	-3	-4	-32
	Rank	5	4	5	4	6	5	4	4	5	6
KnMGPSO_{RI}	Wins	0	0	0	0	0	0	0	0	0	0
	Losses	4	3	4	3	3	4	3	3	4	31
	Difference	-4	-3	-4	-3	-3	-4	-3	-3	-4	-31
	Rank	5	4	5	4	5	5	4	4	5	5

Table 6.17: IGD Ranking for 5-objective WFG

Algorithm	**Result**	**5-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
PMGPSO_{STD}	Wins	0	3	3	1	2	4	3	3	3	22
	Losses	0	0	1	0	0	0	0	0	0	1
	Difference	0	3	2	1	2	4	3	3	3	21
	Rank	1	1	3	1	1	1	1	1	1	1
PMGPSO_{R}	Wins	0	3	4	0	2	3	3	3	3	21
	Losses	0	0	0	0	0	0	0	0	0	0
	Difference	0	3	4	0	2	3	3	3	3	21
	Rank	1	1	1	2	1	2	1	1	1	1
PMGPSO_{RI}	Wins	0	3	3	0	0	3	3	3	3	18
	Losses	0	0	0	0	0	1	0	0	0	1
	Difference	0	3	3	0	0	2	3	3	3	17
	Rank	1	1	2	2	3	3	1	1	1	3
KnMGPSO_{STD}	Wins	0	0	1	0	0	2	0	0	0	3
	Losses	0	3	3	1	0	3	3	3	3	19
	Difference	0	-3	-2	-1	0	-1	-3	-3	-3	-16
	Rank	1	4	4	6	3	4	4	4	4	4
KnMGPSO_{R}	Wins	0	0	0	0	0	0	0	0	0	0
	Losses	0	3	4	0	2	4	3	3	3	22
	Difference	0	-3	-4	0	-2	-4	-3	-3	-3	-22
	Rank	1	4	6	2	5	5	4	4	4	6
KnMGPSO_{RI}	Wins	0	0	0	0	0	0	0	0	0	0
	Losses	0	3	3	0	2	4	3	3	3	21
	Difference	0	-3	-3	0	-2	-4	-3	-3	-3	-21
	Rank	1	4	5	2	5	5	4	4	4	5

Table 6.18: IGD Ranking for 8-objective WFG

Algorithm	Result	8-objective WFG 1	2	3	4	5	6	7	8	9	Overall
$PMGPSO_{STD}$	Wins	0	3	3	0	0	3	3	3	3	18
	Losses	4	0	2	0	0	0	0	0	0	6
	Difference	-4	3	1	0	0	3	3	3	3	12
	Rank	6	1	3	1	1	1	1	1	1	3
$PMGPSO_{R}$	Wins	1	3	4	0	0	3	3	3	3	20
	Losses	3	0	0	0	0	0	0	0	0	3
	Difference	-2	3	4	0	0	3	3	3	3	17
	Rank	4	1	1	1	1	1	1	1	1	1
$PMGPSO_{RI}$	Wins	0	3	4	0	0	3	3	3	3	19
	Losses	3	0	0	0	0	0	0	0	0	3
	Difference	-3	3	4	0	0	3	3	3	3	16
	Rank	5	1	1	1	1	1	1	1	1	2
$KnMGPSO_{STD}$	Wins	3	1	0	0	0	0	0	0	1	5
	Losses	0	3	5	0	0	3	3	3	3	20
	Difference	3	-2	-5	0	0	-3	-3	-3	-2	-15
	Rank	1	4	6	1	1	4	4	4	4	5
$KnMGPSO_{R}$	Wins	3	0	1	0	0	0	0	0	0	4
	Losses	0	5	3	0	0	3	3	3	4	21
	Difference	3	-5	-2	0	0	-3	-3	-3	-4	-17
	Rank	1	6	4	1	1	4	4	4	6	6
$KnMGPSO_{RI}$	Wins	3	1	1	0	0	0	0	0	0	5
	Losses	0	3	3	0	0	3	3	3	3	18
	Difference	3	-2	-2	0	0	-3	-3	-3	-3	-13
	Rank	1	4	4	1	1	4	4	4	5	4

Table 6.19: IGD Ranking for 10-objective WFG

Algorithm	**Result**	**10-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
PMGPSO$_{STD}$	Wins	0	3	3	0	3	0	1	3	3	16
	Losses	5	0	1	0	0	0	0	0	0	6
	Difference	-5	3	2	0	3	0	1	3	3	10
	Rank	6	1	3	1	1	3	3	1	1	3
PMGPSO$_{R}$	Wins	1	3	3	0	3	1	2	3	3	19
	Losses	3	0	0	0	0	0	0	0	0	3
	Difference	-2	3	3	0	3	1	2	3	3	16
	Rank	4	1	2	1	1	2	1	1	1	2
PMGPSO$_{RI}$	Wins	1	3	4	0	3	3	2	3	3	22
	Losses	3	0	0	0	0	0	0	0	0	3
	Difference	-2	3	4	0	3	3	2	3	3	19
	Rank	4	1	1	1	1	1	1	1	1	1
KnMGPSO$_{STD}$	Wins	3	2	0	0	0	0	0	0	0	5
	Losses	2	3	3	0	3	1	0	3	3	18
	Difference	1	-1	-3	0	-3	-1	0	-3	-3	-13
	Rank	3	4	4	1	4	4	4	4	4	4
KnMGPSO$_{R}$	Wins	4	0	0	0	0	0	0	0	0	4
	Losses	0	4	3	0	3	2	3	3	3	21
	Difference	4	-4	-3	0	-3	-2	-3	-3	-3	-17
	Rank	1	5	4	1	4	6	6	4	4	6
KnMGPSO$_{RI}$	Wins	4	0	0	0	0	0	0	0	0	4
	Losses	0	4	3	0	3	1	2	3	3	19
	Difference	4	-4	-3	0	-3	-1	-2	-3	-3	-15
	Rank	1	5	4	1	4	4	5	4	4	5

Table 6.20: IGD Ranking for 15-objective WFG

Algorithm	**Result**	**15-objective WFG**									**Overall**
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
$PMGPSO_{STD}$	Wins	0	3	3	3	3	0	0	0	3	15
	Losses	3	0	0	0	0	0	0	0	0	3
	Difference	-3	3	3	3	3	0	0	0	3	12
	Rank	4	1	1	1	1	1	1	1	1	1
$PMGPSO_{R}$	Wins	0	3	3	3	3	0	0	0	3	15
	Losses	3	0	0	0	0	0	0	0	0	3
	Difference	-3	3	3	3	3	0	0	0	3	12
	Rank	4	1	1	1	1	1	1	1	1	1
$PMGPSO_{RI}$	Wins	0	3	3	3	3	0	0	0	3	15
	Losses	3	0	0	0	0	0	0	0	0	3
	Difference	-3	3	3	3	3	0	0	0	3	12
	Rank	4	1	1	1	1	1	1	1	1	1
$KnMGPSO_{STD}$	Wins	3	0	0	1	0	0	0	0	0	4
	Losses	0	3	3	3	3	0	0	0	3	15
	Difference	3	-3	-3	-2	-3	0	0	0	-3	-11
	Rank	1	4	4	4	4	1	1	1	4	4
$KnMGPSO_{R}$	Wins	3	0	0	0	0	0	0	0	0	3
	Losses	0	3	3	4	3	0	0	0	3	16
	Difference	3	-3	-3	-4	-3	0	0	0	-3	-13
	Rank	1	4	4	6	4	1	1	1	4	6
$KnMGPSO_{RI}$	Wins	3	0	0	0	0	0	0	0	0	3
	Losses	0	3	3	3	3	0	0	0	3	15
	Difference	3	-3	-3	-3	-3	0	0	0	-3	-12
	Rank	1	4	4	5	4	1	1	1	4	5

6.2.1 Hypervolume Discussion

The overall HV rankings for each algorithm, shown in tables 6.1 to 6.10, are analyzed and discussed next.

The PMGPSO$_{STD}$ consistently ranked fourth overall in terms of HV (tables 6.1 to 6.10). The PMGPSO$_{STD}$ is the only algorithm that never ranked worst overall (considering both HV and IGD). Therefore, the overall performance of the PMGPSO$_{STD}$ with respect to HV was not competitive although consistent.

The best overall rank obtained by the PMGPSO$_R$ in terms of HV was five, for the 10- and 15-objective DTLZ problems (tables 6.4 and 6.5). The PMGPSO$_R$ performed worst overall in terms of HV the most; that is, eight times for the 3-, 5-, and 8-objective DTLZ problems (tables 6.1 to 6.3) and for the 3-, 5-, 8-, 10-, and 15-objective WFG problems (tables 6.6 to 6.10). Therefore, the PMGPSO$_R$ performed worst overall in terms of HV.

The best overall rank obtained by the PMGPSO$_{RI}$ in terms of HV was five, for the 8-, 10-, and 15-objective WFG problems (tables 6.8 to 6.10). The PMGPSO$_{RI}$ ranked worst overall in terms of HV the second-most; that is, seven times for the rest of the problems (tables 6.1 to 6.7). Therefore, the PMGPSO$_{RI}$ did not perform well in terms of HV. The overall performance of the PMGPSO$_R$ and PMGSO$_{RI}$ was comparable with respect to HV.

The KnMGPSO$_{STD}$ ranked first overall eight times with respect to HV; that is, the most compared to any algorithm. The KnMGPSO$_{STD}$ ranked first overall for the 3-, 5-, 8-, 10-, and 15-objective DTLZ problems (tables 6.1 to 6.5) and for the 8-, 10-, and 15-objective WFG problems (tables 6.8 to 6.10). The KnMGPSO$_{STD}$ never ranked worse than top-three overall with respect to HV. Therefore, the KnMGPSO$_{STD}$ performed best overall with respect to HV.

The KnMGPSO$_R$ ranked best overall only twice with respect to HV; i.e., the least by any algorithm. The KnMGPSO$_R$ ranked first overall for the 3-objective DTLZ problems (table 6.1) and the 5-objective WFG problems (table 6.7). The KnMGPSO$_R$ never ranked worse than top-three overall with respect to HV. Therefore, the KnMGPSO$_R$ performed somewhat competitively in terms of HV.

The KnMGPSO$_{RI}$, like the KnMGPSO$_R$, ranked best overall only twice in terms of HV. The KnMGPSO$_{RI}$ ranked best overall in terms of HV for the 3- and 5-objective WFG problems (tables 6.6 and 6.7). The KnMGPSO$_{RI}$ never ranked worse than top-three overall with respect to HV. Therefore, the KnMGPSO$_{RI}$ performed somewhat competitively in terms of HV. The overall performance of the KnMGPSO$_R$ and KnMGSO$_{RI}$ was comparable with respect to HV.

6.2.2 Inverted Generational Distance Discussion

The overall IGD rankings for each algorithm, shown in tables 6.11 to 6.20, are analyzed and discussed next.

The PMPSO$_{STD}$ performed best overall four times with respect to IGD, which were for the 8-objective DTLZ problems (table 6.13) and the 3-, 5-, and 15-objective WFG problems (tables 6.16, 6.17, and 6.20). The PMGPSO$_{STD}$ never ranked worse than top-three overall with respect to IGD. Therefore, the PMGPSO$_{STD}$ performed competitively in terms of IGD.

The PMGPSO$_{R}$ ranked best overall seven times with respect to IGD; that is, the most by any algorithm. The PMGPSO$_{R}$ ranked first overall for the 3-, 8-, 10-, and 15-objective DTLZ problems (tables 6.11 and 6.13 to 6.15) and also for the 5-, 8-, and 15-objective WFG problems (tables 6.17, 6.18, and 6.20). The PMGPSO$_{R}$ never ranked worse than top-three overall with respect to IGD. Therefore, the PMGPSO$_{R}$ performed best in terms of IGD.

The PMGPSO$_{RI}$ ranked best overall five times with regards to IGD; that is, for the 5- and 15-objective DTLZ problems (tables 6.12 and 6.15) and the 3-, 10-, and 15-objective WFG problems (tables 6.16, 6.19, and 6.20). The PMGPSO$_{RI}$ never ranked worse than top-three overall with respect to IGD. Therefore, the PMGPSO$_{RI}$ performed very competitively in terms of IGD.

The best overall rank obtained by the KnMGPSO$_{STD}$ in terms of IGD was four, for the 8-objective DTLZ problems (table 6.13) and the 3-, 5-, 10-, and 15-objective WFG problems (tables 6.16, 6.17, 6.19, and 6.20). The KnMGPSO$_{STD}$ ranked worst overall four times in terms of IGD, for the 3-, 5-, 10-, and 15-objective DTLZ problems (tables 6.11, 6.12, 6.14, and 6.15). Therefore, the KnMGPSO$_{STD}$ did not perform well in terms of IGD.

The best overall rank obtained by the KnMGPSO$_{R}$ with respect to IGD was four, for the 3-objective DTLZ problems (table 6.11). The KnMGPSO$_{R}$ ranked worst overall the most in terms of IGD. That is, six times for the 8-objective DTLZ problems (table 6.13) and for the 3-, 5-, 8-, 10-, and 15-objective WFG problems (tables 6.16 to 6.20). Therefore, the KnMGPSO$_{R}$ performed worst overall in terms of IGD.

The best overall rank obtained by the KnMGPSO$_{RI}$ in terms of IGD was four, for the 3-, 5-, 10-, and 15-objective DTLZ problems (tables 6.11, 6.12, 6.14, and 6.15) and the 8-objective WFG problems (table 6.18). The KnMGPSO$_{RI}$ ranked worst overall only once in terms of IGD; i.e., for the 8-objective DTLZ problems (table 6.13). Therefore, the KnMGPSO$_{RI}$ did not perform well in terms of IGD. The KnMGPSO$_{RI}$ did, however, outperform both the KnMGPSO$_{STD}$ and the KnMGPSO$_{STD}$ with respect to IGD.

6.2.3 General Discussion

Some general findings with regards to tables 6.1 to 6.20 are discussed next.

The KnMGPSO$_{STD}$, the KnMGPSO$_{R}$, and the KnMGPSO$_{RI}$ ranked top-three overall for each problem with respect to HV (tables 6.1 to 6.10). That is, the knee-points approach (the KnMGPSO algorithm) resulted in higher HV values than the partial-dominance approach (the PMGPSO algorithm). This was expected since Zhang *et al.* [174] noted that the incorporation of knee-points during optimization results in a bias towards a higher HV. Therefore, the KnMGPSO algorithms outperformed the PMGPSO algorithms in terms of HV.

The PMGPSO$_{STD}$, the PMGPSO$_{R}$, and the PMGPSO$_{RI}$ ranked top-three overall for each problem in terms of IGD (tables 6.11 to 6.20). Therefore, the PMGPSO algorithm variants outperformed the KnMGPSO algorithm variants with respect to IGD.

The PMGPSO algorithms dominated in terms of IGD while the KnMGPSO algorithms dominated in terms of HV. Future research will investigate whether it makes sense to combine partial-dominance and knee-points in some way.

The dynamic archive balance coefficient update strategies did not significantly improve algorithm performance. Only the performance of the PMGPSO algorithm was improved slightly with regards to IGD. That is, the PMGPSO$_{R}$ and the PMGPSO$_{RI}$ outperformed the PMGPSO$_{STD}$ with respect to IGD. The PMGPSO$_{R}$ was superior to both the PMGPSO$_{STD}$ and the PMGPSO$_{RI}$ in terms of IGD.

The results, discussed in the sections above, are summarized in tables 6.21 and 6.22. The tables contain the number of overall best, overall top-three, and overall worst ranks obtained by each algorithm with respect to either HV or IGD. Tables 6.23 and 6.24 contain the average overall rank obtained by each algorithm for each benchmark problem suite for each number of objectives with respect to either HV or IGD.

Table 6.21: HV Ranking Summary

Algorithm	**Number of overall best HV ranks**	**Number of overall HV ranks ≤ 3**	**Number of overall worst HV ranks**
PMGPSO$_{STD}$	0	0	0
PMGPSO$_{R}$	0	0	8
PMGPSO$_{RI}$	0	0	7
KnMGPSO$_{STD}$	8	10	0
KnMGPSO$_{R}$	2	10	0
KnMGPSO$_{RI}$	2	10	0

Table 6.22: IGD Ranking Summary

Algorithm	**Number of overall best IGD ranks**	**Number of overall IGD ranks ≤ 3**	**Number of overall worst IGD ranks**
PMGPSO_{STD}	4	10	0
PMGPSO_{R}	7	10	0
PMGPSO_{RI}	5	10	0
KnMGPSO_{STD}	0	0	4
KnMGPSO_{R}	0	0	6
KnMGPSO_{RI}	0	0	1

6.3 Summary

This chapter empirically investigated if either of the two multi-guide particle swarm optimization (MGPSO) algorithm [137] adaptations, the partial-dominance multi-guide particle swarm optimization (PMGPSO) algorithm or the knee-point driven multi-guide particle swarm optimization (KnMGPSO) algorithm, outperforms the other; that is, to determine which of the two proposed scalability-improving mechanisms, if any, is superior to the other. Furthermore, different archive balance coefficient update strategies were investigated to determine if scalability to many-objective optimization (MaOO) is improved by any of the approaches. Three different archive balance coefficient update strategies were also investigated with the aim of improving scalability. These included the standard static archive balance coefficient update strategy (STD), the random dynamic archive balance coefficient update strategy (R), and the random per particle dynamic archive balance coefficient update strategy (RI) [53]. The KnMGPSO and PMGPSO algorithms were tested on a set of benchmark algorithms by calculating, and statistically analyzing, the inverted generational distance (IGD) [25, 128] and hypervolume (HV) [181] performance measure values that were calculated on the normalized solutions without outliers. Problems with 3, 5, 8, 10, and 15 objectives were used to test the scalability of the algorithms. The results were presented in tables and discussed.

The PMGPSO algorithm using the STD (PMGPSO_{STD}) performed very competitively since it, at worst, ranked fourth overall. The PMGPSO algorithm using the R (PMGPSO_{R}) performed best overall in terms of IGD while ranking worst overall most often with respect to HV. The PMGPSO algorithm using the RI (PMGPSO_{RI}) performed competitively in terms of IGD but not in terms of HV. The PMGPSO_{R} and the PMGPSO_{RI} had similar performance.

The KnMGPSO algorithm using the STD (KnMGPSO_{STD}) performed best overall in terms of HV. The KnMGPSO algorithm using the R (KnMGPSO_{R}) performed worst overall since it rarely ranked best overall in terms of HV and ranked worst overall most often in terms of IGD. The KnMGPSO algorithm

Table 6.23: HV Ranking Averages

Algorithm	**Benchmark Problems**	n_m **3**	**5**	**8**	**10**	**15**	**Average**
$PMGPSO_{STD}$	DTLZ	4	4	4	4	4	4
	WFG	4	4	4	4	4	4
	Average Rank	4	4	4	4	4	4
$PMGPSO_{R}$	DTLZ	5	5	5	5	5	5
	WFG	4	5	6	6	6	5.4
	Average Rank	4.5	5	5.5	5.5	5.5	5.2
$PMGPSO_{RI}$	DTLZ	5	5	5	6	6	5.4
	WFG	4	5	5	5	5	4.8
	Average Rank	4.5	5	5	5.5	5.5	5.1
$KnMGPSO_{STD}$	DTLZ	1	1	1	1	1	1
	WFG	3	3	1	1	1	1.8
	Average Rank	2	2	1	1	1	1.4
$KnMGPSO_{R}$	DTLZ	1	3	2	2	2	2
	WFG	2	1	2	2	2	1.8
	Average Rank	1.5	2	2	2	2	1.9
$KnMGPSO_{RI}$	DTLZ	3	2	3	3	2	2.6
	WFG	1	1	2	3	3	2
	Average Rank	2	1.5	2.5	3	2.5	2.3

Table 6.24: IGD Ranking Averages

Algorithm	Benchmark Problems	n_m 3	5	8	10	15	Average
PMGPSO_{STD}	DTLZ	3	3	1	3	2	2.4
	WFG	1	1	3	3	1	1.8
	Average Rank	2	2	2	3	1.5	2.1
PMGPSO_{R}	DTLZ	1	2	1	1	1	1.2
	WFG	3	1	1	2	1	1.6
	Average Rank	2	1.5	1	1.5	1	1.4
PMGPSO_{RI}	DTLZ	2	1	3	2	1	1.8
	WFG	1	3	2	1	1	1.6
	Average Rank	1.5	2	2.5	1.5	1	1.7
KnMGPSO_{STD}	DTLZ	6	6	4	6	6	5.6
	WFG	4	4	5	4	4	4.2
	Average Rank	5	5	4.5	5	5	4.9
KnMGPSO_{R}	DTLZ	4	4	5	5	5	4.6
	WFG	6	6	6	6	6	6
	Average Rank	5	5	5.5	5.5	5.5	5.3
KnMGPSO_{RI}	DTLZ	4	4	5	4	4	4.2
	WFG	5	5	4	5	5	4.8
	Average Rank	4.5	4.5	4.5	4.5	4.5	4.5

using the RI (KnMGPSO$_{RI}$) only performed somewhat competitively in terms of HV.

Note that the PMGPSO algorithms dominated in terms of IGD while the KnMGPSO algorithms dominated in terms of HV. That is, the PMGPSO algorithm variants never ranked top-three overall in terms of HV while the KnMGPSO algorithm variants never ranked top-three overall in terms of IGD.

The dynamic archive balance coefficient update strategies marginally improved the performance of the PMGPSO algorithm with regards to IGD only.

Chapter 7

Conclusions

"The only person you are destined to become is the person you decide to be."
— Ralph W. Emerson

This chapter concludes the dissertation. Section 7.1 summarizes the findings of the thesis and section 7.2 discusses avenues of future work.

7.1 Summary

This study aimed to investigate the ability of the multi-guide particle swarm optimization (MGPSO) algorithm to solve many-objective optimization problems (MaOPs). This work also proposed to investigate different techniques which could be used to help the MGPSO algorithm scale to many-objectives. The proposed techniques included the partial-dominance approach and the knee-points approach, which led to the proposal and development of the partial-dominance multi-guide particle swarm optimization (PMGPSO) algorithm and the knee-point driven multi-guide particle swarm optimization (KnMGPSO) algorithm. The PMGPSO algorithm and the KnMGPSO algorithm are MGPSO algorithm adaptations with the goal of investigating if these approaches will result in better scalability. The Pareto-dominance relation degrades as the number of objectives of the problem continues to increase. It seems that the MGPSO algorithm is not that much affected by this problem. This study also aimed to investigate the use of different archive balance coefficient update strategies for the purpose of aiding the scalability of the MGPSO algorithm. These included the standard static archive balance coefficient update strategy (STD), the random dynamic archive balance coefficient update strategy (R), and the random per particle dynamic archive balance coefficient update strategy (RI).

The results indicated that the MGPSO algorithm using the STD (MGPSO$_{STD}$) scaled competitively considering the hypervolume (HV) rankings in chapters 4 and 5. The PMGPSO algorithm using the STD

($PMGPSO_{STD}$) scaled competitively in terms of the inverted generational distance (IGD) rankings presented in Chapter 4. The KnMGPSO algorithm using the STD ($KnMGPSO_{STD}$) scaled competitively in terms of both the HV and IGD rankings presented in Chapter 5. The partial-dominance approach and knee-points approach did not improve the scalability of the MGPSO algorithm significantly in all cases. Note, however, that the MGPSO, the PMGPSO, and the KnMGPSO algorithms all performed and scaled competitively relative to the state-of-the-art many-objective optimization (MaOO) benchmark algorithms used in this study (chapters 4 and 5). The benchmark algorithms included the controlling dominance area of solutions speed constraint multi-objective particle swarm optimization (CDAS-SMPSO) algorithm, the knee-point driven evolutionary algorithm (KnEA), the many-objective evolutionary algorithm based on dominance and decomposition (MOEA/DD), and the reference-point based non-dominated sorting genetic algorithm (NSGA-III). The NSGA-III scaled well and outperformed all algorithms with respect to IGD in Chapter 5. This can probably be attributed to the reference-points utilized by the NSGA-III to help guide the search.

Note that it is suspected that the multi-swarm approach employed by the MGPSO, the PMGPSO, and the KnMGPSO algorithms significantly improves the performance of these algorithms. This is because each subswarm exploits information mainly with regards to its assigned objective, thereby, increasing the selection pressure. This may have helped to guide the search despite the weaknesses associated with the use of the Pareto-dominance relation (used by the MGPSO and the KnMGPSO algorithms). It may have also helped guide the search even with the stochastic partial-dominance relation (used by the PMGPSO algorithm). Also, the Pareto-dominance relation is only used to update the archive (for the MGPSO and KnMGPSO algorithms) and not to select the personal best, neighbourhood best, or archive guides.

The dynamic archive coefficient update strategies did not improve algorithm performance. That is, the R and the RI did not improve the scalability of the MGPSO, the PMGPSO, or the KnMGPSO algorithms.

In Chapter 6 the PMGPSO algorithm outperformed the KnMGPSO algorithm in terms of IGD but the KnMGPSO algorithm outperformed the PMGPSO algorithm in terms of HV.

Therefore, the MGPSO algorithm scaled well to MaOPs compared to other state-of-the-art MaOO algorithms. The PMGPSO and the KnMGPSO algorithms also scaled competitively, but did not supersede the performance of the MGPSO algorithm indefinitely. The dynamic archive balance coefficient update strategies did not improve scalability for the MGPSO, the PMGPSO, and the KnMGPSO algorithms.

None of the $MGPSO_{STD}$, $PMGPSO_{STD}$, or $KnMGPSO_{STD}$ algorithms was superior to the others in all cases. Therefore, future research is required to better balance the exploration and exploitation during the search process of the MGPSO, PMGPSO, and KnMGPSO algorithms, hopefully resulting in even larger HV and even smaller IGD values. That is, improving the MGPSO algorithm to such a degree that it performs and scales well to many-objectives both in terms of HV and IGD.

7.2 Future Research

Throughout this study, several new ideas for future research have been identified. A summary of each of these ideas is given below.

Convergence- and Diversity Related Mechanisms
Experiment with different combinations of convergence- and diversity-related mechanisms, together with visualizing the archive, with the goal to investigate the diversity and accuracy profile of the MGPSO algorithm (or other variants thereof). For example, consider using weighted distance instead of crowding distance for diversity preservation/promotion (i.e. to promote exploration) [174]. That is, investigate whether using weighted distance for archive deletion and/or archive guide selection would improve the performance of the MGPSO algorithm (and the PMGPSO and KnMGPSO algorithms). Note, however, that the weighted distance approach has the drawback of introducing a parameter for the k-nearest neighbours [56] component, that ideally needs to be tuned since it determines how many of the closest solutions, in objective space, should be considered for the weighted distance calculation. Also, investigate if knee-points are affected by the curse of dimensionality the same way as other issues listed in Section 2.2.2.

Tuning Control Parameters
Investigate the performance difference in the MGPSO algorithm (and any variants thereof) which uses tuned control parameter values instead of resampling new control parameter values throughout the search (as in this study). Also, consider using subswarm specific control parameter values.

Larger Archive Sizes
Investigate larger archive sizes with the aim of improving algorithm performance without unjustifiably increasing the execution time of an algorithm. Storage requirements should also be considered.

Larger Intervals for Resampling Control Parameter Values
Investigate larger sampling intervals as in [67]; i.e., resample control parameter values from the convergent regions of the MGPSO algorithm after a number of iterations instead of each velocity update. It can also be a good idea to resample new control parameter values from the convergent regions only if no performance differences are observed over a period of time. This raises the question of which performance measure to use.

Larger Partial-dominance Objective Sampling Interval
Investigate selecting objectives for the partial-dominance relation after a number of iterations instead of at each archive insert attempt. Larger objective sampling intervals may enhance algorithm performance [52, 69, 133].

Performance-based Partial-dominance
Use roulette wheel selection to determine the objectives selected for the partial-dominance relation. The probability of an objective being selected is based on some performance information. So, if an objective was selected and it resulted in an improvement of performance, then increase the selection probability for that objective, while the others will have to reduce (since the sum of probabilities has to be equal to 1). So the probability of selection is the performance measure value of the selected objective divided by the sum of the performance measure values for the other objectives. This assumes maximization. This raises the question of which performance measure to use.

Hybrid Algorithms
While individual techniques have worked in improving algorithm performance, the current trend is to develop hybrid algorithms, since no one paradigm or mechanism is superior to the others in all situations. In doing so, the respective strengths of different techniques can be utilized while eliminating the weaknesses of individual techniques. Investigate the strengths of the NSGA-III and incorporate the responsible mechanisms into the MGPSO, the PMGPSO, and/or the PMGPSO algorithms. The mechanisms used by the NSGA-III is worth investigating further since it was the only benchmark algorithm that consistently stood out in terms of performance. That is, maybe a MGPSO algorithm variant making use of a reference-front (reference-points) to better guide the search, like the NSGA-III, would scale better to MaOPs. Note, however, that this approach will increase the computational cost tremendously [106]. Also, consider possibly combining the MGPSO_{STD} and the PMGPSO_{STD} in some way, because the former performed best with respect to HV and the latter with respect to IGD. Finally, investigate whether it makes sense to combine partial-dominance and knee-points in some way.

Bibliography

[1] Adra, S.F., Dodd, T.J., Fleming, P.J. and Griffin, I.A., Convergence acceleration operator for multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 825–847, 2009.

[2] Agrawal, R. and Deb, K., Simulated binary crossover for continuous search space. *Complex Systems*, vol. 9, no. 3, 2000.

[3] Auger, A., Bader, J., Brockhoff, D. and Zitzler, E., Theory of the hypervolume indicator: Optimal μ-distributions and the choice of the reference point. *Proceedings of the 10th ACM Special Interest Group on Genetic and Evolutionary Computation Workshop on Foundations of Genetic Algorithms*, pp. 87–102. ACM, 2009.

[4] Bader, J. and Zitzler, E., HypE: An algorithm for fast hypervolume-based many-objective optimization. *Evolutionary Computation*, vol. 19, no. 1, pp. 45–76, 2011.

[5] Barrera, J. and Coello Coello, C.A., A particle swarm optimization method for multimodal optimization based on electrostatic interaction. *Proceedings of the 8th Mexican International Conference on Artificial Intelligence*, vol. 5845, pp. 622–632. Springer, 2009.

[6] Bartz-Beielstein, T., Parsopoulos, K. and Vrahatis, M., Tuning PSO parameters through sensitivity analysis. *University of Dortmund*, 2002.

[7] Bauer, D.F., Constructing confidence sets using rank statistics. *Journal of the American Statistical Association*, vol. 67, no. 339, pp. 687–690, 1972.

[8] Bechikh, S., Ghédira, K. and Ben Said, L., Searching for knee regions in multi-objective optimization using mobile reference points. *Proceedings of the ACM Symposium on Applied Computing*, pp. 1118–1125. ACM, 2010.

[9] Beume, N., S-metric calculation by considering dominated hypervolume as Klee's measure problem. *Evolutionary Computation*, vol. 17, no. 4, pp. 477–492, 2009.

[10] Beume, N., Naujoks, B. and Emmerich, M., SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, 2007.

[11] Brankem, J., Deb, K., Dierolf, H. and Osswald, M., Finding knees in multi-objective optimization. *Proceedings of the 8th International Conference on Parallel Problem Solving from Nature*, vol. 3242, pp. 722–731. 2004.

[12] Bremermann, H.J., Optimization through evolution and recombination. *Self-organizing Systems*, pp. 93–106, 1962.

[13] Bringmann, K. and Friedrich, T., An efficient algorithm for computing hyper-volume contributions. *Evolutionary Computation*, vol. 18, no. 3, pp. 383–402, 2010.

[14] Brockhoff, D. and Zitzler, E., Are all objectives necessary? on dimensionality reduction in evolutionary multiobjective optimization. *Proceedings of the 9th International Conference on Parallel Problem Solving from Nature*, pp. 533–542. Springer, 2006.

[15] Brockhoff, D. and Zitzler, E., Objective reduction in evolutionary multiobjective optimization: Theory and applications. *Evolutionary Computation*, vol. 17, no. 2, pp. 135–166, 2009.

[16] Chelouah, R. and Baron, C., Ant colony algorithm hybridized with tabu and greedy searches as applied to multi-objective optimization in project management. *Journal of Heuristics*, vol. 13, no. 6, p. 640, 2007.

[17] Cleghorn, C.W., Particle swarm optimization: Understanding order-2 stability guarantees. *Proceedings of Applications of Evolutionary Computation*, pp. 535–549. Springer International Publishing, 2019.

[18] Cleghorn, C.W. and Engelbrecht, A.P., A generalized theoretical deterministic particle swarm model. *Swarm Intelligence*, vol. 8, no. 1, pp. 35–59, 2013.

[19] Cleghorn, C.W. and Engelbrecht, A.P., Particle swarm convergence: An empirical investigation. *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 1, pp. 2524–2530. IEEE, 2014.

[20] Cleghorn, C.W., Engelbrecht, A.P. and Scheepers, C., Stability analysis of the multi-objective multi-guided particle swarm optimizer. *Proceedings of the 11th International ANTS Conference*, pp. 201–212. Springer, 2018.

[21] Clerc, M., The swarm and the queen: towards a deterministic and adaptive particle swarm optimization. *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 3, pp. 1951–1957. IEEE, 1999.

[22] Clerc, M., Think locally, act locally: The way of life of Cheap-PSO, an adaptive PSO. Tech. Rep., 2001.
Available at: `http://clerc.maurice.free.fr/pso/`

[23] Clerc, M. and Kennedy, J., The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.

[24] Cloete, T., Engelbrecht, A.P. and Pampará, G., CIlib: A collaborative framework for computational intelligence algorithms - part ii. *Proceedings of the IEEE International Joint Conference on Neural Networks*, pp. 1764–1773. 2008.

[25] Coello Coello, C.A. and Reyes Sierra, M., A study of the parallelization of a coevolutionary multi-objective evolutionary algorithm. Monroy, R., Arroyo-Figueroa, G., Sucar, L.E. and Sossa, H. (editors), *Proceedings of the Mexican International Conference on Artificial Intelligence*, vol. 2972, pp. 688–697. Springer, 2004.

[26] Coello Coello, C.A., van Veldhuizen, D.A. and Lamont, G., *Evolutionary Algorithms for Solving Multi-Objective Problems*. 2nd edition. Kluwer Academic, 2007.

[27] Dantzig, G.B. and Thapa., M.N., *Linear Programming 2: Theory and Extensions*. 1st edition. Springer, 2003.

[28] Darwin, C., *On the origin of species by means of natural selection, or, The preservation of favoured races in the struggle for life*. 1st edition. London: John Murray, 1859.

[29] Das, I., On characterizing the "knee" of the pareto curve based on normal-boundary intersection. *Structural Optimization*, vol. 18, no. 2, pp. 107–115, 1999.

[30] Das, I. and Dennis, J., A closer look at drawbacks of minimizing weighted sums of objectives for pareto set generation in multicriteria optimization problems. *Structural Optimization*, vol. 14, no. 1, pp. 63–69, 1997.

[31] Das, I. and Dennis, J., Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems. *Society for Industrial and Applied Mathematics Journal on Optimization*, vol. 8, no. 3, pp. 631–657, 2000.

[32] de Carvalho, A.B. and Pozo, A., The control of dominance area in particle swarm optimization algorithms for many-objective problems. *Proceedings of the 11th Brazilian Symposium on Neural Networks*, pp. 140–145. IEEE, 2010.

[33] de Carvalho, A.B. and Pozo, A., Measuring the convergence and diversity of CDAS multi-objective particle swarm optimization algorithms: A study of many-objective problems. *Neurocomputing*, vol. 75, no. 1, pp. 43–51, 2012.

[34] Deb, K., Multi-objective genetic algorithms: Problem difficulties and construction of test problems. *Evolutionary Computation*, vol. 7, no. 3, pp. 205–230, 1999.

[35] Deb, K., Agarwal, S., Meyarivan, T. and Pratap, A., A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[36] Deb, K. and Goyal, M., A combined genetic adaptive search (GeneAS) for engineering design. *Computer Science and Informatics*, vol. 26, no. 4, pp. 30–45, 1996.

[37] Deb, K. and Gupta, S., Understanding knee points in bicriteria problems and their implications as preferred solution principles. *Engineering Optimization*, vol. 43, no. 11, pp. 1175–1204, 2011.

[38] Deb, K. and Jain, H., An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.

[39] Deb, K., Korhonen, P.J., Sinha, A. and Wallenius, J., An interactive evolutionary multiobjective optimization method based on progressively approximated value functions. *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 5, pp. 723–739, 2010.

[40] Deb, K. and Sundar, J., Reference point based multi-objective optimization using evolutionary algorithms. *Proceedings of the Conference on Genetic and Evolutionary Computation*, pp. 635–642. ACM, 2006.

[41] Deb, K., Thiele, L., Laumanns, M. and Zitzler, E., *Scalable Test Problems for Evolutionary Multiobjective Optimization*, pp. 105–145. Springer, 2005.

[42] Demsar, J., Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, vol. 7, no. 1, pp. 1–30, 2006.

[43] Derrac, J., García, S., Molina, D. and Herrera, F., A practical tutorial on the use of nonprametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithm. *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.

[44] di Pierro, F., Savic, D.A. and Khu, S.-T., An investigation on preference order ranking scheme for multiobjective evolutionary optimization. *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 1, pp. 17–45, 2007.

[45] Eberhart, R., Dobbins, R. and Simpson, P., *Computational Intelligence PC Tools*. Academic Press Professional, Inc., 1996.

[46] Eberhart, R. and Kennedy, J., A new optimizer using particle swarm theory. *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pp. 39–43. IEEE, 1995.

[47] Engelbrecht, A.P., *Computational Intelligence: An Introduction*. 2nd edition. Wiley Publishing, 2007.

[48] Engelbrecht, A.P., Particle swarm optimization: Velocity initialization. *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1–8. 2012.

[49] Engelbrecht, A.P., Particle swarm optimization: Global best or local best? *Proceedings of the BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence*, pp. 124–135. IEEE, 2013.

[50] Engelbrecht, A.P., Roaming behavior of unconstrained particles. *Proceedings of the 2013 BRICS Congress on Computational Intelligence & 11th Brazilian Congress on Computational Intelligence*, pp. 104–111. IEEE Computer Society, 2013.

[51] Engelbrecht, A.P., Fitness function evaluations: A fair stopping condition? *Proceedings of the IEEE Symposium on Swarm Intelligence*, pp. 1–8. 2014.

[52] Engelbrecht, A.P. and Helbig, M., Empirical analysis of a partial dominance approach to many-objective optimisation. *Proceedings of the International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence*. 2020.

[53] Erwin, K. and Engelbrecht, A.P., Control parameter sensitivity analysis of the multi-guide particle swarm optimization algorithm. *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 22–29. ACM, 2019.

[54] Ester, M., Kriegel, H.-P., Sander, J. and Xu, X., A density-based algorithm for discovering clusters in large spatial databases with noise. *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, pp. 226–231. AAAI Press, 1996.

[55] Farina, M. and Amato, P., A fuzzy definition of "optimality" for many-criteria optimization problems. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 34, no. 3, pp. 315–326, 2004.

[56] Fix, E. and Hodges., J.L., Discriminatory analysis, nonparametric discrimination: Consistency properties. Tech. Rep., US Air Force School of Aviation Medicine, 1951.

[57] Fleischer, M., The measure of pareto optima applications to multi-objective metaheuristics. *Proceedings of the 2nd International Conference on Evolutionary Multi-Criterion Optimization*, pp. 519–533. Springer, 2003.

[58] Fonseca, C.M., Lopez-Ibanez, M. and Paquete, L., An improved dimension-sweep algorithm for the hypervolume indicator. *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1157–1163. IEEE, 2006.

[59] Friedman, M., The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1937.

[60] Friedman, M., A comparison of alternative tests of significance for the problem of m rankings. *The Annals of Mathematical Statistics*, vol. 11, no. 1, pp. 86–92, 1940.

[61] García, S., Fernández, A., Herrera, F. and Luengo, J., Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Information Sciences*, vol. 180, no. 10, pp. 2044–2064, 2010.

[62] Gazi, V., Stochastic stability analysis of the particle dynamics in the PSO algorithm. *Proceedings of the IEEE International Symposium on Intelligent Control*, pp. 708–713. IEEE, 2012.

[63] Goh, C.-K. and Tan, K., An investigation on noisy environments in evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 3, pp. 354–381, 2007.

[64] Hadka, D. and Reed, P., Borg: An auto-adaptive many-objective evolutionary computing framework. *Evolutionary Computation*, vol. 21, no. 2, pp. 231–259, 2013.

[65] Harrison, K.R., Engelbrecht, A.P. and Ombuki-Berman, B.M., Knowledge transfer strategies for vector evaluated particle swarm optimization. *Proceedings of the 7th International Conference on Evolutionary Multi-Criterion Optimization*, vol. 7, pp. 171–184. 2013.

[66] Harrison, K.R., Engelbrecht, A.P. and Ombuki-Berman, B.M., The sad state of self-adaptive particle swarm optimizers. *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 431–439. IEEE, 2016.

[67] Harrison, K.R., Engelbrecht, A.P. and Ombuki-Berman, B.M., An adaptive particle swarm optimization algorithm based on optimal parameter regions. *Proceedings of the IEEE Symposium on Swarm Intelligence*, pp. 1–8. 2017.

[68] He, Z., Yen, G.G. and Zhang, J., Fuzzy-based pareto optimality for many-objective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 2, pp. 269–285, 2014.

[69] Helbig, M. and Engelbrecht, A.P., Partial dominance for many-objective optimization. *Proceedings of the International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence*. 2020.

[70] Helwig, S. and Wanka, R., Theoretical analysis of initial particle swarm behavior. *Proceedings of the 10th International Conference on Parallel Problem Solving from Nature*, vol. 5199, pp. 889–898. Springer, 2008.

[71] Holland, J.H., *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. MIT Press, 1992.

[72] Hollander, M. and Wolfe, D., *Nonparametric statistical methods*. Wiley Series in Probability and Statistics - Applied Probability and Statistics Section. Wiley, 1973.

[73] Holm, S., A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, vol. 6, no. 2, pp. 65–70, 1979.

[74] Huband, S., Barone, L., Hingston, P. and While, L., A scalable multi-objective test problem toolkit. Coello Coello, C.A., Hernández Aguirre, A. and Zitzler, E. (editors), *Proceedings of the 3rd International Conference on Evolutionary Multi-Criterion Optimization*, pp. 280–295. Springer, 2005.

[75] Huband, S., Barone, L., Hingston, P. and While, L., A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 5, pp. 477–506, 2006.

[76] Huiyuan, F., A modification to particle swarm optimization algorithm. *Engineering Computations*, vol. 19, no. 8, pp. 970–989, 2002.

[77] Ikeda, K., Kita, H. and Kobayashi, S., Failure of pareto-based MOEAs: does non-dominated really mean near to optimal? *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 2, pp. 957–962. IEEE, 2001.

[78] Iman, R.L. and Davenport, J.M., Approximations of the critical region of the Friedman statistic. *Communications in Statistics - Theory and Methods*, vol. 9, no. 6, pp. 571–595, 1980.

[79] Ishibuchi, H., Hitotsuyanagi, Y., Nojima, Y. and Tsukamoto, N., Many-objective test problems to visually examine the behavior of multiobjective evolution in a decision space. Schaefer, R., Cotta, C., Kołodziej, J. and Rudolph, G. (editors), *Proceedings of the 11th International Conference on Parallel Problem Solving from Nature: Part II*, vol. 6239, pp. 91–100. Springer, 2010.

[80] Ishibuchi, H., Masuda, H., Nojima, Y. and Tanigaki, Y., Difficulties in specifying reference points to calculate the inverted generational distance for many-objective optimization problems. *Proceedings of the IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making*, pp. 170–177. IEEE, 2014.

[81] Ishibuchi, H., Nojima, Y. and Tsukamoto, N., Evolutionary many-objective optimization: A short review. *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 2419–2426. IEEE, 2008.

[82] Jiang, M., Luo, Y.P. and Yang, S.Y., Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm. *Information Processing Letters*, vol. 102, no. 1, pp. 8–16, 2007.

[83] Jin, Y., Olhofer, M. and Sendhoff, B., Dynamic weighted aggregation for evolutionary multi-objective optimization: Why does it work and how? *Proceedings of the Conference on Genetic and Evolutionary Computation*, pp. 1042–1049. Morgan Kaufmann Publishers Inc., 2001.

[84] Kadirkamanathan, V., Selvarajah, K. and Fleming, P.J., Stability analysis of the particle dynamics in particle swarm optimizer. *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 3, pp. 245–255, 2006.

[85] Kennedy, J., The particle swarm: social adaptation of knowledge. *Proceedings of IEEE Congress on Evolutionary Computation*, pp. 303–308. IEEE, 1997.

[86] Kennedy, J. and Eberhart, R.C., Particle swarm optimization. *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948. IEEE, 1995.

[87] Kennedy, J. and Eberhart, R.C., *Swarm Intelligence.* Morgan Kaufmann Publishers Inc., 2001.

[88] Kirkpatrick, S., Gelatt, C.D. and Vecchi, M.P., Optimization by simulated annealing. *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[89] Kitayama, S., Srirat, J., Arakawa, M. and Yamazaki, K., Sequential approximate multi-objective optimization using radial basis function network. *Structural and Multidisciplinary Optimization*, vol. 48, no. 3, pp. 501–515, 2013.

[90] Köppen, M. and Yoshida, K., Substitute distance assignments in NSGA-II for handling many-objective optimization problems. *Proceedings of the 4th International Conference on Evolutionary Multi-Criterion Optimization*, pp. 727–741. Springer, 2007.

[91] Kukkonen, S. and Deb, K., Improved pruning of non-dominated solutions based on crowding distance for bi-objective optimization problems. *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1179–1186. IEEE, 2006.

[92] Lakhdar, N., Arar, D., Bendib, T., Djeffal, F. and Meguellati, M., An optimised submicron dual-material gate (DM) GaAs-MESFETs design to improve the analog performance using multi-objective computation. *Proceedings of the 5th International Conference on Modeling, Simulation and Applied Optimization*, vol. 12, pp. 1–5. IEEE, 2013.

[93] Laumanns, M., Deb, K., Thiele, L. and Zitzler, E., Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary Computation*, vol. 10, no. 3, pp. 263–282, 2002.

[94] Li, H. and Zhang, Q., Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II. *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 284–302, 2009.

[95] Li, K., Deb, K., Kwong, S. and Zhang, Q., An evolutionary many-objective optimization algorithm based on dominance and decomposition. *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 5, pp. 694–716, 2015.

[96] Li, M., Li, K., Shen, R., Yuan, Q. and Zheng, J., A grid-based fitness strategy for evolutionary many-objective optimization. *Proceedings of the Conference on Genetic and Evolutionary Computation*, pp. 463–470. ACM, 2010.

[97] Li, X., Adaptively choosing neighbourhood bests using species in a particle swarm optimizer for multimodal function optimization. Deb, K. (ed.), *Proceedings of the Genetic and Evolutionary Computation Conference*, vol. 3102, pp. 105–116. Springer, 2004.

[98] Li, X., A multimodal particle swarm optimizer based on fitness euclidean-distance ratio. *Proceedings of the Conference on Genetic and Evolutionary Computation*, pp. 78–85. ACM, 2007.

[99] Lindroth, P., Patriksson, M. and Strömberg, A.-B., Approximating the pareto optimal set using a reduced set of objective functions. *European Journal of Operational Research*, vol. 207, no. 3, pp. 1519–1534, 12 2010.

[100] Liu, F.T., Ting, K.M. and Zhou, Z., Isolation forest. *Proceedings of the 8th IEEE International Conference on Data Mining*, pp. 413–422. IEEE, 2008.

[101] Liu, F.T., Ting, K.M. and Zhou, Z.-H., Isolation-based anomaly detection. *ACM Transactions on Knowledge Discovery from Data*, vol. 6, no. 1, 2012.

[102] López Jaimes, A., Coello, C., Fujii, K. and Oyama, A., An alternative preference relation to deal with many-objective optimization problems. *Proceedings of the 7th International Conference Evolutionary Multi-criterion Optimization*, pp. 291–306. 2013.

[103] López Jaimes, A., Coello Coello, C.A. and Chakraborty, D., Objective reduction using a feature selection technique. *Proceedings of the Conference on Genetic and Evolutionary Computation*, pp. 673–680. ACM, 2008.

[104] Lygoe, R.J., Cary, M. and Fleming, P.J., A real-world application of a many-objective optimisation complexity reduction process. Purshouse, R., Fleming, P., Fonseca, C., Greco, S. and Shaw, J. (editors), *Proceedings of the International Conference on Evolutionary Multi-Criterion Optimization*, vol. 7811, pp. 641–655. 2013.

[105] Maltese, J., Engelbrecht, A.P. and Ombuki-Berman, B.M., Pareto-based many-objective optimization using knee points. *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 3678–3686. 2016.

[106] Maltese, J., Engelbrecht, A.P. and Ombuki-Berman, B.M., A Scalability Study of Many-Objective Optimization Algorithms. *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 21, pp. 79–96, 2018.

[107] Mane, S. and Rao, M., Many-objective optimization: Problems and evolutionary algorithms - a short review. *International Journal of Applied Engineering Research*, vol. 12, no. 20, pp. 9774–9793, 2017.

[108] Mendes, R., Wanner, E., Martins, F. and Sarubbi, J.a., Dimensionality reduction approach for many-objective vehicle routing problem with demand responsive transport. *Proceedings of the 9th International Conference on Evolutionary Multi-Criterion Optimization*, vol. 10173, pp. 438–452. Springer, 2017.

[109] Molina, J., Caballero, R., Coello, C., Hernández-Díaz, A. and Santana-Quintero, L., G-dominance: Reference point based dominance for multiobjective metaheuristics. *European Journal of Operational Research*, vol. 197, no. 2, pp. 685–692, 2009.

[110] Naka, S., Genji, T., Yura, T. and Fukuyama, Y., Practical distribution state estimation using hybrid particle swarm optimization. *Proceedings of the IEEE Power Engineering Society Winter Conference*, vol. 2, pp. 815–820. 2001.

[111] Narukawa, K. and Rodemann, T., Examining the performance of evolutionary many-objective optimization algorithms on a real-world application. *Proceedings of the 6th International Conference on Genetic and Evolutionary Computing*, pp. 316–319. 2012.

[112] Nebro, A.J., Alba, E., Coello, C.A.C., Durillo, J.J., Garcia-Nieto, J. and Luna, F., SMPSO: A new PSO-based metaheuristic for multi-objective optimization. *Proceedings of the IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making*, pp. 66–73. IEEE, 2009.

[113] Otten, R.H.J.M. and van Ginneken, L.P.P.P., *The Annealing Algorithm*. Kluwer, 1989.

[114] Özcan, E. and Mohan, C., Analysis of a simple particle swarm optimization system. *Proceedings of the Conference on Intelligent Engineering Systems Through Artificial Neural Networks*, vol. 8, pp. 253–258. 1998.

[115] Özcan, E. and Mohan, C.K., Particle swarm optimization: surfing the waves. *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 3, pp. 1939–1944. IEEE, 1999.

[116] Pampará, G., Engelbrecht, A.P. and Cloete, T., CIlib: A collaborative framework for computational intelligence algorithms - part i. *Proceedings of the IEEE International Joint Conference on Neural Networks*, pp. 1750–1757. 2008.

[117] Parsopoulos, K. and Vrahatis, M., Recent approaches to global optimization problems through particle swarm optimization. *Natural Computing*, vol. 116, no. 2-3, pp. 235–306, 2002.

[118] Parsopoulos, K.E. and Vrahatis, M.N., Particle swarm optimization method in multiobjective problems. *Proceedings of the ACM Symposium on Applied Computing*, pp. 603–607. ACM, 2002.

[119] Peng, J., Chen, Y. and Eberhart, R., Battery pack state of charge estimator design using computational intelligence approaches. *Proceedings of the Battery Conference on Applications and Advances*, pp. 173–177. 2000.

[120] Peram, T., Mohan, C.K. and Veeramachaneni, K., Fitness-distance-ratio based particle swarm optimization. *Proceedings of the IEEE Swarm Intelligence Symposium*, pp. 174–181. 2003.

[121] Poli, R., Mean and variance of the sampling distribution of particle swarm optimizers during stagnation. *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 712–721, 2009.

[122] Poli, R. and Broomhead, D., Exact analysis of the sampling distribution for the canonical particle swarm optimiser and its convergence during stagnation. *Proceedings of the Conference on Genetic and Evolutionary Computation*, pp. 134–141. ACM, 2007.

[123] Prel, J.-B., Röhrig, B., Hommel, G. and Blettner, M., Choosing statistical tests - part 12 of a series on evaluation of scientific publications. *Deutsches Ärzteblatt International*, vol. 107, no. 19, pp. 343–348, 2010.

[124] Rachmawati, L. and Srinivasan, D., A multi-objective evolutionary algorithm with weighted-sum niching for convergence on knee regions. *Proceedings of the 8th Conference on Genetic and Evolutionary Computation*, pp. 749–750. 2006.

[125] Rachmawati, L. and Srinivasan, D., A multi-objective genetic algorithm with controllable convergence on knee regions. *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1916–1923. 2006.

[126] Rashid, M., A., B. and Zafar, K., Niching with sub-swarm based particle swarm optimization. *Proceedings of the International Conference on Computer Technology and Development*, vol. 2, pp. 181–183. 2009.

[127] Ratnaweera, A., Halgamuge, S. and Watson, H., Particle swarm optimization with self-adaptive acceleration coefficients. *Proceedings of the First International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 264–268. 2003.

[128] Reyes Sierra, M. and Coello Coello, C.A., A new multi-objective particle swarm optimizer with improved selection and diversity mechanisms. Tech. Rep., Evolutionary Computation Group, 2004.

[129] Reyes Sierra, M. and Coello Coello, C.A., Improving PSO-based multi-objective optimization using crowding, mutation and ϵ-dominance. *Lecture Notes in Computer Science*, vol. 3410, pp. 505–519, 2005.

[130] Said, L.B., Bechikh, S. and Ghédira, K., The R-dominance: A new dominance relation for interactive evolutionary multicriteria decision making. *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 5, pp. 801–818, 2010.

[131] Sato, H., Aguirre, H. and Tanaka, K., Self-controlling dominance area of solutions in evolutionary many-objective optimization. *Proceedings of Simulated Evolution and Learning*, vol. 6457, pp. 455–465. 2010.

[132] Sato, H., Aguirre, H.E. and Tanaka, K., Controlling dominance area of solutions and its impact on the performance of MOEAs. Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T. and Murata, T. (editors), *Proceedings of the 4th International Conference on Evolutionary Multi-Criterion Optimization*, pp. 5–20. Springer, 2007.

[133] Sato, H., Aguirre, H.E. and Tanaka, K., Pareto partial dominance MOEA and hybrid archiving strategy included CDAS in many-objective optimization. *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1–8. IEEE, 2010.

[134] Saxena, D.K. and Deb, K., Dimensionality reduction of objectives and constraints in multi-objective optimization problems: A system design perspective. *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 3204–3211. 2008.

[135] Schaffer, J.D., Multiple objective optimization with vector evaluated genetic algorithms. *Proceedings of the 1st International Conference on Genetic Algorithms*, pp. 93–100. L. Erlbaum Associates Inc., 1985.

[136] Scheepers, C., *Multi-guided Particle Swarm Optimization A Multi-swarm Multi-objective Particle Swarm Optimizer.* Ph.D. thesis, University of Pretoria, 2017.

[137] Scheepers, C., Cleghorn, C.W. and Engelbrecht, A.P., Multi-guide particle swarm optimization for multi-objective optimization: empirical and stability analysis. *Swarm Intelligence*, vol. 13, no. 2–3, pp. 245–276, 2019.

[138] Schott, J.R., *Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization.* Master's thesis, Massachusetts Institute of Technology, 1995.

[139] Schubert, E., Sander, J., Ester, M., Kriegel, H.P. and Xu, X., DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN. *ACM Transactions on Database Systems*, vol. 42, no. 3, 2017.

[140] Schutte, J.F. and Groenwold, A., Sizing design of truss structures using particle swarms. *Structural and Multidisciplinary Optimization*, vol. 25, no. 4, pp. 261–269, 2003.

[141] Schutze, O., Coello Coello, C.A. and Lara, A., On the influence of the number of objectives on the hardness of a multiobjective optimization problem. *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 4, pp. 444–455, 2011.

[142] Schütze, O., Laumanns, M. and Coello, C.A., Approximating the knee of an MOP with stochastic search algorithms. *Proceedings of the 10th International Conference on Parallel Problem Solving from Nature*, vol. 5199, pp. 795–804. Springer, 2008.

[143] Sheskin, D.J., *Handbook of Parametric and Nonparametric Statistical Procedures.* 4th edition. Chapman & Hall/CRC, 2007.

[144] Shi, Y. and Eberhart, R., A modified particle swarm optimizer. *Proceedings of the IEEE Congress on Evolutionary Computation Proceedings and IEEE World Congress on Computational Intelligence*, vol. 6, pp. 69–73. IEEE, 1998.

[145] Shi, Y. and Eberhart, R.C., Parameter selection in particle swarm optimization. Porto, V.W., Saravanan, N., Waagen, D. and Eiben, A.E. (editors), *Proceedings of the 7th International Conference on Evolutionary Programming VII*, vol. 1447, pp. 591–600. Springer, 1998.

[146] Shi, Y. and Eberhart, R.C., Empirical study of particle swarm optimization. *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 3, pp. 1945–1950. IEEE, 1999.

[147] Shi, Y. and Eberhart, R.C., Fuzzy adaptive particle swarm optimization. *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 1, pp. 101–106. 2001.

[148] Singh, H.K., Isaacs, A. and Ray, T., A pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems. *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 4, pp. 539–556, 2011.

[149] Suganthan, P., Particle swarm optimiser with neighborhood operator. *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 3, pp. 1958–1962. IEEE, 1999.

[150] Sülflow, A., Drechsler, N. and Drechsler, R., Robust multi-objective optimization in high dimensional spaces. Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T. and Murata, T. (editors), *Proceedings of the 4th International Conference on Evolutionary Multi-Criterion Optimization*, vol. 4403, pp. 715–726. Springer, 2007.

[151] Tan, K.C., Khor, E.F. and Lee, T.H., *Multiobjective Evolutionary Algorithms and Applications (Advanced Information and Knowledge Processing)*. Springer, 2005.

[152] Thiele, L., Miettinen, K., Molina, J. and Korhonen, P.J., A preference-based evolutionary algorithm for multi-objective optimization. *Evolutionary Computation*, vol. 17, no. 3, pp. 411–436, 2009.

[153] Tian, Y., Cheng, R., Jin, Y. and Zhang, X., PlatEMO: A MATLAB platform for evolutionary multi-objective optimization. *IEEE Computational Intelligence Magazine*, vol. 12, no. 4, pp. 73–87, 2017.

[154] Trelea, I.C., The particle swarm optimization algorithm: convergence analysis and parameter selection. *Information Processing Letters*, vol. 85, no. 6, pp. 317–325, 2003.

[155] Tušar, T. and Filipič, B., Scaling and visualizing multiobjective optimization test problems with knees. *Proceedings of the 15th International Multiconference Information Society*, pp. 155–158. 2012.

[156] Upton, G. and Cook, I., *Understanding Statistics*. OUP Oxford, 1996.

[157] van den Bergh, F., *An Analysis of Particle Swarm Optimizers*. Ph.D. thesis, University of Pretoria, 2002.

[158] van den Bergh, F. and Engelbrecht, A.P., A study of particle swarm optimization particle trajectories. *Information Sciences*, vol. 176, no. 8, pp. 937–971, 2006.

[159] van Veldhuizen, D.A., *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations*. Ph.D. thesis, Air Force Institute of Technology, 1999.

[160] van Veldhuizen, D.A. and Lamont, G.B., *Multiobjective Evolutionary Algorithm Research: A History and Analysis*. Ph.D. thesis, Air Force Institute of Technology, 1998.

[161] van Veldhuizen, D.A. and Lamont, G.B., On measuring multiobjective evolutionary algorithm performance. *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 1, pp. 204–211. IEEE, 2000.

[162] van Zyl, E. and Engelbrecht, A.P., Comparison of self-adaptive particle swarm optimizers. *Proceedings of the IEEE Symposium on Swarm Intelligence*, pp. 1–9. IEEE, 2014.

[163] Venter, G. and Sobieszczanski-Sobieski, J., Multidisciplinary optimization of a transport aircraft wing using particle swarm optimization. *Structural and Multidisciplinary Optimization*, vol. 26, no. 1–2, pp. 121–131, 2003.

[164] Venter, G. and Sobieszczanski-Sobieski, J., Particle swarm optimization. *Journal for the American Institute of Aeronautics and Astronautics*, vol. 41, no. 8, pp. 1583–1589, 2003.

[165] Wagner, T., Beume, N. and Naujoks, B., Pareto-, aggregation-, and indicator-based methods in many-objective optimization. *Proceedings of the 4th International Conference on Evolutionary Multi-Criterion Optimization*, pp. 742–756. Springer, 2007.

[166] Wang, G. and Jiang, H., Fuzzy-dominance and its application in evolutionary many objective optimization. *Proceedings of the International Conference on Computational Intelligence and Security Workshops*, pp. 195–198. IEEE, 2007.

[167] While, L., Barone, L. and Bradstreet, L., A fast way of calculating exact hypervolumes. *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 1, pp. 86–95, 2012.

[168] While, L., Hingston, P., Barone, L. and Huband, S., A faster algorithm for calculating hypervolume. *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 1, pp. 29–38, 2006.

[169] Wilcoxon, F., Individual comparisons by ranking methods. *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.

[170] Yasuda, K., Ide, A. and Iwasaki, N., Adaptive particle swarm optimization. *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics.*, vol. 2, pp. 1554–1559. IEEE, 2003.

[171] Yen, G.G. and He, Z., Performance metric ensemble for multiobjective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 1, pp. 131–144, 2014.

[172] Yuan, Y. and Xu, H., Multiobjective flexible job shop scheduling using memetic algorithms. *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 1, pp. 336–353, 2015.

[173] Zhang, Q. and Li, H., MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.

[174] Zhang, X., Jin, Y. and Tian, Y., A knee point-driven evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 6, pp. 761–776, 2015.

[175] Zhao, W.Z., Chen, T., Wang, C.Y. and Yu, L., Performance optimization of electric power steering based on multi-objective genetic algorithm. *Journal of Central South University*, vol. 20, no. 1, pp. 98–104, 2013.

[176] Zheng, Y.-L., Ji-Xin, Q., Li-Yan, Z. and Long-Hua, M., On the convergence analysis and parameter selection in particle swarm optimization. *Proceedings of the International Conference on Machine Learning and Cybernetics*, vol. 3, pp. 1802–1807. IEEE, 2003.

[177] Zitzler, E., *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. Ph.D. thesis, Swiss Federal Institute of Technology Zurich, 1999.

[178] Zitzler, E., Deb, K. and Thiele, L., Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation*, vol. 8, no. 2, pp. 173–195, 2000.

[179] Zitzler, E. and Künzli, S., Indicator-based selection in multiobjective search. *Proceedings of the 8th International Conference on Parallel Problem Solving from Nature*, pp. 832–842. 2004.

[180] Zitzler, E. and Thiele, L., Multiobjective optimization using evolutionary algorithms - a comparative case study. *Proceedings of the 5th International Conference on Parallel Problem Solving from Nature*, vol. 1498, pp. 292–301. Springer, 1998.

[181] Zitzler, E. and Thiele, L., Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.

[182] Zou, X., Chen, Y., Kang, L. and Liu, M., A new evolutionary algorithm for solving many-objective optimization problems. *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, vol. 38, no. 5, pp. 1402–1412, 2008.

[183] Zwillinger, D. and Kokoska, S., *CRC Standard Probability and Statistics Tables and Formulae, Student Edition*. 1st edition. CRC Press, 2000.

Appendices

Appendix A

Acronyms

This appendix lists the acronyms that were used throughout this study. Acronyms are typeset in bold, with the corresponding meaning alongside. The acronyms are listed in alphabetical order.

AI Artificial Intelligence.

ANOVA Analysis of Variance.

CDAS Controlling Dominance Area of Solutions.

CDAS-SMPSO Controlling Dominance Area of Solutions Speed Constraint Multi-objective Particle Swarm Optimization.

CHPC Centre for High Performance Computing.

CI Computational Intelligence.

CIlib Computational Intelligence library.

COVID-19 Coronavirus disease.

CSIR Council for Scientific and Industrial Research.

CSP Sonstraint Satisfaction Problem.

DTLZ Deb-Thiele-Laumanns-Zitzler.

EA Evolutionary Algorithm.

EC Evolutionary Computation.

GA Genetic Algorithm.

gbest global best.

HV Hypervolume.

IGD Inverted Generational Distance.

IQR Interquartile Range.

KnEA Knee-point driven Evolutionary Algorithm.

KnMGPSO Knee-point driven Multi-guide Particle Swarm Optimization.

KnMGPSO$_{R}$ Knee-point driven Multi-guide Particle Swarm Optimization algorithm with the Random dynamic archive balance coefficient update strategy.

KnMGPSO$_{RI}$ Knee-point driven Multi-guide Particle Swarm Optimization algorithm with the Random per Particle dynamic archive balance coefficient update strategy.

KnMGPSO$_{STD}$ Knee-point driven Multi-guide Particle Swarm Optimization algorithm with the standard static archive balance coefficient update strategy.

lbest local best.

LD Linearly Decreasing dynamic archive balance coefficient update strategy.

LI Linearly Increasing dynamic archive balance coefficient update strategy.

MaOEA Many-objective Evolutionary Algorithm.

MaOO Many-objective Optimization.

MaOP Many-objective Optimization Problem.

MaOPSO Many-objective Particle Swarm Optimization.

MGPSO Multi-guide Particle Swarm Optimization.

MGPSO$_{R}$ Multi-guide Particle Swarm Optimization algorithm with the random dynamic archive balance coefficient update strategy.

MGPSO$_{RI}$ Multi-guide particle swarm optimization algorithm with the random per particle dynamic archive balance coefficient update strategy.

MGPSO$_{STD}$ Multi-guide Particle Swarm Optimization algorithm with the standard static archive balance coefficient update strategy.

MOEA Multi-objective Evolutionary Algorithm.

MOEA/D Multi-objective Evolutionary Algorithm based on Decomposition.

MOEA/DD Many-objective Evolutionary Algorithm based on Dominance and Decomposition.

MOO Multi-objective Optimization.

MOP Multi-objective Optimization Problem.

MOPSO Multi-objective Particle Swarm Optimization.

NSGA-II Non-dominated Sorting Genetic Algorithm.

NSGA-III Reference-point based many-objective Non-dominated Sorting Genetic Algorithm.

PBI Penalty-based Boundary Intersection.

PlatEMO Platform for Evolutionary Multi-objective Optimization.

PMGPSO Partial-dominance Multi-guide Particle Swarm Optimization.

PMGPSO$_{R}$ Partial-dominance Multi-guide Particle Swarm Optimization algorithm with the Random dynamic archive balance coefficient update strategy.

PMGPSO$_{RI}$ Partial-dominance Multi-guide Particle Swarm Optimization algorithm with the Random per Particle dynamic archive balance coefficient update strategy.

PMGPSO$_{STD}$ Partial-dominance Multi-guide Particle Swarm Optimization algorithm with the standard static archive balance coefficient update strategy.

POF Pareto-optimal Front.

POS Pareto-optimal Set.

PSO Particle Swarm Optimization.

R Random dynamic archive balance coefficient update strategy.

RI Random per article dynamic archive balance coefficient update strategy.

SBX Simulated Binary Crossover.

SI Swarm Intelligence.

SMPSO Speed Constraint Multi-objective Particle Swarm Optimization.

STD Standard static archive balance coefficient update strategy.

UOO Uni-objective Optimization.

VEPSO Vector Evaluated Particle Swarm Optimization.

WFG Walking Fish Group.

Appendix B

Symbols

This appendix lists the important symbols used throughout the dissertation, as well as their corresponding meanings.

$\mathcal{A}$	set of archive solutions
$\hat{\mathbf{a}}_i(t)$	archive guide for the i-th particle at iteration t
b_k	Euclidean distance in the objective space between solution k of the true POF and the closest member of the found front
$\mathcal{C}(t)$	population of individuals at generation (iteration) t
c_1	cognitive acceleration coefficient
c_2	social acceleration coefficient
c_3	archive acceleration coefficient
E	list of extremal solutions used to construct matrix $\mathbf{G}$
E_m	least desirable solution for objective m
$\mathcal{F}$	feasible space

f	objective function (to be minimized)
f_m	minimization objective function for objective m
$f_m(\mathbf{x})$	quantity associated with the quality of candidate solution $\mathbf{x}$ for objective m
$\mathbf{f}(\mathbf{x})$	objective vector quantifying the quality of candidate solution $\mathbf{x}$
$f''_m(\mathbf{x})$	CDAS-modified quantity associated with the quality of candidate solution $\mathbf{x}$ for objective m
f^*_m	optimal objective value for objective m
f'	objective function (to be maximized)
f^{**}_m	worst (least desirable) objective value for objective m in the POS
$g^{pbi}(\mathbf{x}\|\mathbf{w}, \mathbf{z}^*)$	quantity associated with the quality of solution $\mathbf{x}$ with regards to the PBI aggregation approach
$\mathbf{G}$	matrix used to determine the constants of the hyperplane H
H	extremal hyperplane
HV	HV performance measure
i	particle index

IGD	IGD performance measure
j, k	generic set component indices
j_n	archive guide coefficient component index
m	objective index
$max_m(t)$	maximum value for objective m at iteration t
$min_m(t)$	minimum value for objective m at iteration t
n	number of decision variables
n_m	number of objectives
n_s	number of particles/individuals (also known as the swarm/population size)
n_{s_m}	subswarm size for objective m
n_t	maximum number of iterations
$\mathcal{O}$	objective space
$\mathcal{P}$	POS
P_m^{PD}	probability of the m-th objective being chosen for the partial-dominance relation
POF^*	set of non-dominated objective vectors representing the true POF

$\mathcal{Q}$	set of non-dominated objective vectors that make up the approximated POF
$ratio(t)$	ratio of the neighbourhood size to the range spanned by obejctive m at iteration t as used by the knee-point identification approach
$\mathbf{r}_{hv}$	reference-point for the HV (hypercube) calculation
$R_m(t)$	neighbourhood of particle for objective m when identifying knee-points
$\mathbb{R}^n$	n-dimensional real number space
$\mathbb{R}^{n_m}$	n_m-dimensional real number space
$\mathbf{r}_1$	first uniform random stochastic vector with components in the range [0, 1]
$\mathbf{r}_2$	second uniform random stochastic vector with components in the range [0, 1]
$\mathbf{r}_3$	third uniform random stochastic vector with components in the range [0, 1]
$\mathcal{S}$	search space
S	swarm of particles
S_m	subswarm of particles associated with (or responsible for) optimizing objective m

t	current iteration/generation (i.e. the current time step)
$\mathbf{v}_i(t)$	velocity of the i-th particle at iteration t
$\mathcal{V}_k$	hypercube formed between solution $\mathbf{f(x)}_k$ and a reference point $\mathbf{r}_{hv}$
$\mathbf{v}_{max}$	maximum particle velocity
$\mathbf{w}$	weight vector (i.e. reference-point)
W	set of weight vectors (i.e. reference-points) used by the MOEA/DD and the NSGA-III
w	inertia weight
$\mathbf{x}$	decision vector (i.e. a candidate solution, particle, or chromosome)
$\mathbf{x}_i(t)$	position (or gene alleles) of particle (or chromosome/individual) i at iteration t
$\mathbf{x}_{max}$	upper bound of the search space when all dimensions have the same boundary
x_{max}	upper bound of the search space in some dimension
$\mathbf{x}_{min}$	lower bound of the search space when all dimensions have the same boundary
x_{min}	lower bound of the search space in some dimension
$\mathbf{y}_i(t)$	personal best position of the i-th particle at iteration t

	(cognitive guide)
$\hat{\mathbf{y}}_i(t)$	neighbourhood best position of the i-th particle at iteration t (social guide)
$\mathbf{z}^{\text{nad}}$	nadir objective vector
$\mathbf{z}^*$	ideal objective vector
χ	constriction factor (coefficient)
δ	neighbourhood selection probability
γ	parameter controlling the contraction and expansion of the CDAS method
κ	desired ratio of knee-points to non-dominated solutions (i.e. the rate of knee-points)
$\lambda_{ij}(t)$	j-th component of the archive balance coefficient for particle i at iteration t
φ	PBI penalty parameter
ψ	MOEA/DD neighbourhood size
$\zeta(t-1)$	ratio of knee-points to non-dominated solutions at iteration $t-1$
$\prec$	Pareto-dominance operator

Appendix C

Parameter Configurations

This appendix provides a list of the parameter configurations for the benchmark algorithms used in this study that required parameter tuning. The experimental work for the below listed algorithms were conducted using the PlatEMO [153] framework with the recommended well-performing parameter configurations found throughout the literature [33, 95, 174]. Tables C.1, C.2, and C.3 lists the parameter configurations for the KnEA, the MOEA/DD, and the CDAS-SMPSO algorithm respectively.

Table C.1: Tuned Parameter Values for the KnEA

KnEA					
κ					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.2	0.3	0.2	0.7	0.2
DTLZ2	0.2	0.8	0.3	0.7	0.6
DTLZ3	0.5	0.1	0.6	0.4	0.6
DTLZ4	0.8	0.6	0.6	0.9	0.7
DTLZ5	0.8	0.5	0.4	0.2	0.6
DTLZ6	0.4	0.3	0.4	0.1	0.3
DTLZ7	0.5	0.8	0.8	0.4	0.6
WFG1	0.7	0.8	0.7	0.4	0.9
WFG2	0.8	0.5	0.3	0.8	0.7
WFG3	0.9	0.8	0.2	0.7	0.1
WFG4	0.7	0.5	0.7	0.5	0.5
WFG5	0.8	0.6	0.7	0.6	0.8
WFG6	0.9	0.6	0.4	0.9	0.9
WFG7	0.6	0.5	0.8	0.6	0.9
WFG8	0.9	0.6	0.7	0.5	0.6
WFG9	0.5	0.6	0.7	0.5	0.2

Table C.2: Tuned Parameter Values for the MOEA/DD

MOEA/DD					
δ					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.2	1.0	1.0	0.9	1.0
DTLZ2	0.0	0.0	0.5	1.0	0.3
DTLZ3	1.0	1.0	0.8	1.0	1.0
DTLZ4	0.1	0.2	0.2	1.0	0.0
DTLZ5	0.1	0.1	0.4	0.5	0.1
DTLZ6	0.1	0.0	0.0	0.2	0.3
DTLZ7	0.1	0.6	0.8	1.0	0.5
WFG1	0.1	0.3	0.0	0.0	0.1
WFG2	0.2	0.6	0.8	1.0	0.7
WFG3	0.5	0.5	0.0	0.2	0.1
WFG4	0.3	0.2	1.0	1.0	0.8
WFG5	0.4	0.2	0.6	0.8	0.7
WFG6	0.8	0.0	0.9	0.1	0.1
WFG7	0.4	0.5	1.0	0.9	0.6
WFG8	0.8	0.2	0.9	0.2	0.5
WFG9	0.6	0.2	0.8	0.3	0.6

Table C.3: Tuned Parameter Values for the CDAS-SMPSO algorithm

CDAS-SMPSO γ					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.25	0.25	0.25	0.25	0.25
DTLZ2	0.35	0.35	0.35	0.35	0.35
DTLZ3	0.35	0.3	0.25	0.35	0.25
DTLZ4	0.5	0.5	0.5	0.5	0.5
DTLZ5	0.4	0.4	0.4	0.4	0.4
DTLZ6	0.25	0.25	0.25	0.25	0.25
DTLZ7	0.35	0.35	0.3	0.25	0.45
WFG1	0.45	0.5	0.5	0.5	0.5
WFG2	0.5	0.5	0.45	0.4	0.35
WFG3	0.45	0.45	0.45	0.45	0.5
WFG4	0.5	0.45	0.6	0.45	0.7
WFG5	0.45	0.45	0.6	0.6	0.65
WFG6	0.45	0.45	0.45	0.45	0.5
WFG7	0.45	0.45	0.4	0.55	0.35
WFG8	0.45	0.4	0.55	0.65	0.4
WFG9	0.45	0.45	0.4	0.35	0.3

Appendix D

Performance Measure Values for Chapter 4

This appendix provides the average, standard deviation, maximum, and minimum HV and IGD performance measure values for each algorithm on each problem instance for Chapter 4. Section D.1 lists the HV performance measure tables; that is, tables D.1 to D.10. Section D.2 lists the IGD performance measure tables; that is, tables D.11 to D.20. Note that the tables are listed alphabetically according to algorithm name. Also, note that some of the algorithms had no valid solutions left over after outlier removal; that is, no solutions in any of the independent samples for that specific problem. In these rare cases no performance measure value could be calculated; indicated with "-".

D.1 Hypervolume Values

The average, standard deviation, maximum, and minimum HV performance measure values for each algorithm on each problem instance are listed in tables D.1 to D.10. Note that these performance measure values are associated with Chapter 4.

Table D.1: Average, Standard Deviation, Maximum, and Minimum HV for the CDAS-SMPSO algorithm

CDAS-SMPSO HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.1357	1.3125	1.4184	1.7371	3.3500
	0.0404	0.0789	0.2067	0.4681	0.4094
	1.2104	1.4477	1.8199	2.4801	3.9672
	1.0485	1.1361	0.9328	0.6701	2.2402
DTLZ2	1.2722	1.5217	1.9481	2.3269	3.3020
	0.0379	0.0414	0.0670	0.0984	0.4377
	1.3123	1.5710	2.0554	2.4873	3.9508
	1.1754	1.4202	1.8313	2.0640	1.7962
DTLZ3	1.2354	1.4085	1.8465	2.2846	3.6132
	0.0992	0.0563	0.0923	0.2133	0.3240
	1.3210	1.6098	1.9731	2.5641	3.8947
	0.8882	1.2793	1.4836	1.7090	2.5868
DTLZ4	1.1509	1.4886	1.8167	1.9809	2.7113
	0.1444	0.1065	0.2468	0.3988	0.5150
	1.3056	1.5794	2.0901	2.5731	3.8472
	0.8166	1.1525	1.0120	1.1566	1.6532
DTLZ5	1.1577	1.2681	1.4657	1.5488	2.2675
	0.0429	0.0898	0.1882	0.1424	0.4562
	1.2594	1.4724	1.8632	1.9724	3.2805
	1.0906	1.1373	1.1665	1.3483	1.3836
DTLZ6	0.9170	1.1089	1.5343	1.9192	3.1058
	0.0879	0.1117	0.1686	0.1511	0.4343
	1.0803	1.3289	1.9037	2.2595	3.9458
	0.7284	0.7986	1.3007	1.6358	2.1760

Table D.1: Average, Standard Deviation, Maximum, and Minimum HV for the CDAS-SMPSO algorithm (continue)

CDAS-SMPSO HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1034	0.0371	0.0024	0.0017	0.0010
	0.1216	0.0674	0.0078	0.0071	0.0033
	0.4162	0.2846	0.0422	0.0391	0.0150
	0.0149	0.0004	0.0000	0.0000	0.0000
WFG1	1.0261	1.1982	0.5775	0.3129	0.0222
	0.0509	0.0372	0.1253	0.1591	0.0104
	1.0662	1.2513	0.8830	0.6180	0.0299
	0.8740	1.0866	0.4500	0.0910	0.0070
WFG2	0.9765	1.0336	0.7076	0.4845	1.8310
	0.0292	0.0432	0.1136	0.2118	0.5736
	1.0457	1.1023	0.8955	0.9290	2.7255
	0.8946	0.9377	0.4079	0.0787	0.3379
WFG3	0.7562	0.7559	0.6802	0.7875	1.8559
	0.0223	0.0192	0.0385	0.0529	0.1238
	0.8054	0.7936	0.7669	0.8868	2.0642
	0.7072	0.7071	0.5951	0.6851	1.5697
WFG4	0.7587	0.9176	1.0273	1.0835	1.6461
	0.0149	0.0415	0.0351	0.1029	0.2800
	0.7855	1.0163	1.1069	1.2940	1.9732
	0.7317	0.8317	0.9501	0.8334	0.7266
WFG5	0.7788	0.8273	0.8895	0.7564	0.6921
	0.0196	0.0330	0.0453	0.0766	0.2514
	0.8185	0.8965	0.9920	0.9165	1.1202
	0.7510	0.7711	0.7851	0.5873	0.2417

Table D.1: Average, Standard Deviation, Maximum, and Minimum HV for the CDAS-SMPSO algorithm (continue)

CDAS-SMPSO HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6799	0.7360	0.7464	0.7236	0.9377
	0.0228	0.0292	0.0393	0.0691	0.2278
	0.7073	0.8115	0.8314	0.9040	1.3590
	0.6137	0.6841	0.6669	0.5433	0.2913
WFG7	0.7867	0.8568	0.8634	0.8297	1.3364
	0.0114	0.0301	0.0338	0.0774	0.1762
	0.8067	0.9048	0.9375	0.9707	1.6016
	0.7585	0.7970	0.7996	0.6852	0.9431
WFG8	0.7720	0.7602	0.7953	0.7530	1.3021
	0.0164	0.0384	0.0345	0.0694	0.1462
	0.8048	0.8095	0.8570	0.8715	1.5906
	0.7359	0.6073	0.7249	0.6006	0.9503
WFG9	0.8318	0.8562	0.9012	0.7711	1.2606
	0.0297	0.0301	0.0742	0.2139	0.2575
	0.8965	0.9174	1.0079	1.0705	1.6945
	0.7629	0.7924	0.6303	0.2060	0.2950

Table D.2: Average, Standard Deviation, Maximum, and Minimum HV for the KnEA

KnEA HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3251	-	2.0701	2.4788	3.9868
	0.0008	-	0.0370	0.0765	0.1232
	1.3263	-	2.1100	2.5748	4.1392
	1.3228	-	1.9093	2.2513	3.6886
DTLZ2	1.3095	1.5607	1.6904	2.1901	2.9177
	0.0019	0.0564	0.2248	0.2301	0.4846
	1.3131	1.6054	1.9870	2.5321	3.9176
	1.3053	1.3894	1.2250	1.7060	2.2293
DTLZ3	1.3185	1.4932	1.9345	2.2669	3.8488
	0.0080	0.1271	0.1564	0.2657	0.3018
	1.3233	1.5995	2.0780	2.5387	4.1481
	1.2774	1.1485	1.5562	1.2509	3.2682
DTLZ4	1.2817	1.3204	1.4430	1.7968	2.4787
	0.0395	0.1604	0.0373	0.0376	0.0726
	1.3050	1.5600	1.4949	1.8511	2.5925
	1.0987	1.1274	1.3467	1.6619	2.3423
DTLZ5	1.1629	1.0150	1.0997	1.1107	1.6922
	0.0946	0.1041	0.0697	0.2082	0.6308
	1.2423	1.4775	1.2032	2.0671	3.2977
	1.0051	0.8742	0.9409	0.7964	1.1245
DTLZ6	1.0278	0.6209	0.7052	0.9619	1.5910
	0.0168	0.2092	0.2062	0.1957	0.4548
	1.0509	1.2337	1.1836	1.5486	2.6475
	0.9981	0.4179	0.4985	0.7375	1.0583

Table D.2: Average, Standard Deviation, Maximum, and Minimum HV for the KnEA (continue)

KnEA HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.3951	0.8557	0.9364	0.5275	0.1051
	0.0420	0.0410	0.1736	0.1605	0.0580
	0.4286	0.9495	1.2319	0.9730	0.2435
	0.1822	0.7891	0.4949	0.2984	0.0110
WFG1	0.8130	0.9437	0.6856	0.6150	0.0775
	0.0512	0.1861	0.1753	0.1416	0.0222
	1.0381	1.2424	1.0206	0.9050	0.1148
	0.7365	0.6819	0.4454	0.3504	0.0427
WFG2	0.9506	0.9900	0.6980	0.7028	2.0528
	0.0575	0.0699	0.0912	0.1499	0.4985
	1.0178	1.1275	0.9487	1.0158	2.8801
	0.8584	0.8333	0.5122	0.3816	1.0861
WFG3	0.7057	0.6163	0.4554	0.3817	0.4460
	0.0300	0.0366	0.0532	0.1095	0.4268
	0.7505	0.6852	0.5645	0.6600	1.5702
	0.6413	0.5426	0.3471	0.2152	0.0041
WFG4	0.7951	1.0125	0.7828	0.9416	0.7251
	0.0139	0.0361	0.1241	0.1276	0.2874
	0.8177	1.0792	1.0067	1.1554	1.3178
	0.7451	0.9421	0.4328	0.6416	0.2606
WFG5	0.8401	0.9864	0.5197	0.8144	0.6281
	0.0134	0.0398	0.1936	0.1546	0.2719
	0.8777	1.0492	0.8435	1.0776	1.1649
	0.8221	0.8866	0.1988	0.4494	0.1513

Table D.2: Average, Standard Deviation, Maximum, and Minimum HV for the KnEA (continue)

KnEA HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.7217	0.8660	0.4998	0.7077	0.6390
	0.0171	0.0385	0.1529	0.1368	0.2835
	0.7702	0.9322	0.8076	1.0011	1.3947
	0.6972	0.7561	0.1649	0.4762	0.2384
WFG7	0.8039	0.9791	0.6050	0.6274	0.3828
	0.0144	0.0381	0.0938	0.1413	0.2781
	0.8286	1.0566	0.7916	0.9645	1.3027
	0.7748	0.9083	0.4267	0.3582	0.0696
WFG8	0.8155	0.8569	0.5075	0.5221	0.5960
	0.0148	0.0665	0.1248	0.1733	0.4419
	0.8504	0.9934	0.7005	0.8766	1.3803
	0.7707	0.6939	0.2714	0.2176	0.0616
WFG9	0.7968	0.8826	0.6325	0.6942	0.7000
	0.0258	0.0486	0.1114	0.1637	0.4044
	0.8726	1.0315	0.9160	1.0938	1.5249
	0.7449	0.7945	0.4747	0.3405	0.1116

Table D.3: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_R$

MGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3120	1.5399	1.9914	2.3914	3.9004
	0.0024	0.0104	0.0262	0.0283	0.0383
	1.3178	1.5593	2.0262	2.4446	3.9585
	1.3077	1.5037	1.8977	2.3142	3.8224
DTLZ2	1.3214	1.5898	2.0784	2.4433	3.7608
	0.0004	0.0030	0.0108	0.0257	0.0994
	1.3221	1.5939	2.0985	2.4925	3.8928
	1.3205	1.5815	2.0570	2.3705	3.4487
DTLZ3	1.3041	1.5270	1.9767	2.3060	3.7915
	0.0031	0.0077	0.0284	0.0809	0.1079
	1.3096	1.5448	2.0467	2.4202	3.9434
	1.2986	1.5089	1.9280	2.0158	3.5465
DTLZ4	1.1952	1.5218	1.5516	1.7845	2.5076
	0.1108	0.0868	0.1023	0.0480	0.0891
	1.3196	1.5950	1.9887	1.8645	2.6329
	1.0797	1.2898	1.2925	1.6166	2.2881
DTLZ5	1.2711	1.4956	1.9120	2.2422	3.3743
	0.0004	0.0032	0.0039	0.0083	0.0218
	1.2717	1.5007	1.9207	2.2667	3.4323
	1.2702	1.4849	1.9059	2.2292	3.3365
DTLZ6	0.7830	1.0748	1.3382	1.3642	1.9419
	0.0087	0.0119	0.0376	0.0656	0.1403
	0.8015	1.0964	1.4112	1.4675	2.2019
	0.7655	1.0490	1.2697	1.1598	1.5867

Table D.3: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_R$ (continue)

MGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.1781	1.3528	1.2660	1.0182	1.2964
	0.0602	0.0432	0.0559	0.1469	0.2151
	1.2630	1.4229	1.3542	1.3386	1.8038
	1.0445	1.2520	1.1539	0.7741	0.8764
WFG1	0.5362	0.6332	0.8355	0.8244	3.6280
	0.0190	0.0473	0.0984	0.1799	0.2125
	0.5712	0.7932	1.1012	1.3682	3.9559
	0.4937	0.5467	0.6499	0.5679	3.1336
WFG2	0.8646	0.9934	1.2733	1.4948	2.4517
	0.0357	0.0288	0.0203	0.0202	0.0314
	0.8887	1.0317	1.3086	1.5360	2.5012
	0.6855	0.8633	1.2353	1.4546	2.3771
WFG3	0.7849	0.9019	1.1168	1.2520	2.1289
	0.0156	0.0093	0.0225	0.0323	0.0489
	0.8193	0.9200	1.1530	1.3130	2.2033
	0.7487	0.8799	1.0638	1.1943	2.0455
WFG4	0.7662	0.7886	1.0381	1.1726	1.9555
	0.0212	0.0220	0.0180	0.0336	0.0502
	0.8051	0.8346	1.0823	1.2291	2.0686
	0.7212	0.7480	1.0092	1.0992	1.8728
WFG5	0.7524	0.7106	0.7634	0.7711	1.3282
	0.0399	0.0264	0.0237	0.0358	0.1033
	0.8299	0.7586	0.8096	0.8422	1.5138
	0.6864	0.6523	0.7196	0.6996	1.1188

Table D.3: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_R$ (continue)

MGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.7025	0.7192	0.8274	0.8968	1.4641
	0.0151	0.0179	0.0252	0.0292	0.0428
	0.7246	0.7548	0.8713	0.9522	1.5361
	0.6589	0.6790	0.7760	0.8341	1.3609
WFG7	0.5917	0.6298	0.7578	0.8290	1.2827
	0.0109	0.0104	0.0223	0.0233	0.0426
	0.6105	0.6577	0.8260	0.8764	1.3958
	0.5688	0.6107	0.7083	0.7783	1.2151
WFG8	0.5669	0.5922	0.7090	0.8017	1.2452
	0.0067	0.0128	0.0221	0.0265	0.0479
	0.5822	0.6177	0.7484	0.8478	1.3395
	0.5549	0.5675	0.6650	0.7443	1.1484
WFG9	0.9660	1.0022	1.1525	1.2459	1.9361
	0.0188	0.0314	0.0385	0.0656	0.1013
	1.0033	1.1074	1.2521	1.4153	2.0800
	0.9214	0.9464	1.0850	1.1050	1.7334

Table D.4: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{RI}$

MGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3115	1.5397	1.9899	2.3837	3.9012
	0.0024	0.0104	0.0247	0.0331	0.0361
	1.3185	1.5593	2.0283	2.4446	3.9798
	1.3077	1.5037	1.8977	2.2908	3.8013
DTLZ2	1.3214	1.5897	2.0807	2.4538	3.7692
	0.0005	0.0036	0.0111	0.0269	0.1054
	1.3222	1.5966	2.0985	2.5058	3.9215
	1.3198	1.5800	2.0570	2.3705	3.3851
DTLZ3	1.3041	1.5254	1.9817	2.2982	3.8036
	0.0031	0.0100	0.0322	0.0734	0.0953
	1.3096	1.5476	2.0497	2.4202	4.0081
	1.2959	1.4963	1.8698	2.0158	3.5465
DTLZ4	1.1938	1.5240	1.5463	1.7883	2.5188
	0.1086	0.0849	0.1200	0.0437	0.0769
	1.3201	1.5952	1.9887	1.8645	2.6329
	1.0797	1.2898	1.1723	1.6166	2.2881
DTLZ5	1.2711	1.4949	1.9114	2.2428	3.3773
	0.0004	0.0067	0.0052	0.0074	0.0213
	1.2717	1.5007	1.9223	2.2667	3.4323
	1.2702	1.4494	1.8973	2.2292	3.3365
DTLZ6	0.7822	1.0742	1.3336	1.3741	1.9234
	0.0082	0.0115	0.0350	0.0606	0.1339
	0.8015	1.1035	1.4112	1.4941	2.2019
	0.7655	1.0490	1.2465	1.1598	1.5867

Table D.4: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{RI}$ (continue)

MGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.1695	1.3622	1.2684	0.9892	1.2662
	0.0573	0.0402	0.0543	0.1299	0.2230
	1.2630	1.4501	1.3909	1.3386	1.8038
	1.0445	1.2520	1.1539	0.7282	0.7350
WFG1	0.5403	0.6380	0.8548	0.8207	3.6058
	0.0214	0.0510	0.1196	0.2104	0.1981
	0.5816	0.7946	1.3245	1.7379	3.9559
	0.4915	0.5467	0.6499	0.5679	3.1336
WFG2	0.8698	0.9887	1.2757	1.5000	2.4556
	0.0262	0.0458	0.0190	0.0190	0.0322
	0.8912	1.0317	1.3172	1.5360	2.5393
	0.6855	0.7733	1.2215	1.4546	2.3771
WFG3	0.7870	0.8984	1.1179	1.2555	2.1440
	0.0143	0.0112	0.0208	0.0344	0.0536
	0.8225	0.9200	1.1694	1.3402	2.2519
	0.7487	0.8702	1.0638	1.1873	2.0278
WFG4	0.7684	0.7901	1.0402	1.1743	1.9502
	0.0195	0.0210	0.0229	0.0310	0.0509
	0.8051	0.8346	1.0833	1.2462	2.0686
	0.7193	0.7480	0.9902	1.0992	1.8290
WFG5	0.7518	0.7073	0.7675	0.7720	1.3246
	0.0399	0.0268	0.0301	0.0402	0.1065
	0.8299	0.7674	0.8540	0.8808	1.5138
	0.6665	0.6523	0.7196	0.6958	1.0968

Table D.4: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{RI}$ (continue)

MGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.7039	0.7208	0.8278	0.9042	1.4600
	0.0130	0.0191	0.0239	0.0311	0.0497
	0.7274	0.7611	0.8713	0.9708	1.5577
	0.6589	0.6640	0.7760	0.8341	1.3128
WFG7	0.5914	0.6324	0.7561	0.8285	1.2780
	0.0131	0.0142	0.0249	0.0244	0.0445
	0.6210	0.6857	0.8359	0.8836	1.3958
	0.5622	0.6107	0.7006	0.7716	1.1667
WFG8	0.5659	0.5928	0.7096	0.7970	1.2496
	0.0075	0.0127	0.0210	0.0330	0.0492
	0.5842	0.6270	0.7484	0.8478	1.3395
	0.5499	0.5675	0.6650	0.6640	1.1478
WFG9	0.9678	0.9970	1.1450	1.2478	1.9239
	0.0161	0.0279	0.0374	0.0648	0.1041
	1.0033	1.1074	1.2521	1.4153	2.1026
	0.9214	0.9290	1.0850	1.0504	1.6463

Table D.5: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{STD}$

MGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3158	1.5430	1.9980	2.3890	3.9058
	0.0069	0.0128	0.0294	0.0401	0.0408
	1.3297	1.5944	2.1307	2.5063	3.9798
	1.3077	1.5037	1.8977	2.2908	3.7926
DTLZ2	1.3212	1.5895	2.0797	2.4523	3.7611
	0.0007	0.0036	0.0121	0.0313	0.1029
	1.3222	1.5978	2.0993	2.5094	3.9215
	1.3176	1.5800	2.0357	2.3516	3.3273
DTLZ3	1.3080	1.5313	1.9898	2.3022	3.8104
	0.0073	0.0156	0.0338	0.0751	0.0975
	1.3281	1.5999	2.1333	2.4353	4.0459
	1.2959	1.4963	1.8698	2.0158	3.5465
DTLZ4	1.1960	1.5059	1.5827	1.8063	2.5567
	0.1080	0.0945	0.1289	0.0474	0.0847
	1.3201	1.5952	1.9887	1.8968	2.6795
	1.0797	1.2897	1.1723	1.6166	2.2881
DTLZ5	1.2708	1.4949	1.9122	2.2439	3.3787
	0.0006	0.0057	0.0055	0.0107	0.0250
	1.2717	1.5007	1.9290	2.2788	3.4493
	1.2689	1.4494	1.8973	2.2195	3.3212
DTLZ6	0.7795	1.0727	1.3360	1.3835	1.9298
	0.0092	0.0114	0.0330	0.0580	0.1360
	0.8015	1.1035	1.4112	1.5193	2.2019
	0.7598	1.0490	1.2465	1.1598	1.5867

Table D.5: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{STD}$ (continue)

MGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.1619	1.3599	1.2970	1.1094	1.4855
	0.0583	0.0388	0.0797	0.2303	0.4298
	1.2630	1.4501	1.5497	1.8105	2.6837
	1.0442	1.2520	1.0987	0.7282	0.7350
WFG1	0.5273	0.6191	0.7971	0.7520	3.4781
	0.0275	0.0534	0.1322	0.2016	0.5894
	0.5816	0.7946	1.3245	1.7379	3.9559
	0.4716	0.5203	0.5641	0.4836	0.3016
WFG2	0.8664	0.9856	1.2746	1.4952	2.4507
	0.0270	0.0450	0.0175	0.0410	0.0337
	0.8912	1.0317	1.3172	1.5520	2.5393
	0.6855	0.7717	1.2215	1.1828	2.3771
WFG3	0.7821	0.8928	1.1213	1.2532	2.1401
	0.0164	0.0145	0.0236	0.0432	0.0582
	0.8225	0.9200	1.1716	1.3530	2.2519
	0.7450	0.8469	1.0561	1.1516	1.9857
WFG4	0.7664	0.7882	1.0375	1.1668	1.9384
	0.0219	0.0221	0.0241	0.0314	0.0556
	0.8065	0.8416	1.0833	1.2462	2.0686
	0.7151	0.7245	0.9847	1.0992	1.7897
WFG5	0.7584	0.7076	0.7778	0.7888	1.3680
	0.0524	0.0291	0.0338	0.0465	0.1255
	0.9503	0.7975	0.9043	0.9053	1.6660
	0.6665	0.6523	0.7196	0.6958	1.0968

Table D.5: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{STD}$ (continue)

MGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6958	0.7145	0.8221	0.9003	1.4469
	0.0179	0.0214	0.0271	0.0326	0.0537
	0.7274	0.7611	0.8713	0.9976	1.5577
	0.6445	0.6502	0.7496	0.8311	1.2696
WFG7	0.5894	0.6328	0.7595	0.8325	1.2757
	0.0143	0.0160	0.0281	0.0285	0.0469
	0.6229	0.6857	0.8359	0.9133	1.4120
	0.5517	0.6053	0.6997	0.7716	1.1667
WFG8	0.5637	0.5921	0.7123	0.8086	1.2542
	0.0085	0.0128	0.0216	0.0364	0.0512
	0.5842	0.6270	0.7652	0.8819	1.3814
	0.5435	0.5649	0.6650	0.6640	1.1478
WFG9	0.9643	0.9978	1.1420	1.2450	1.9195
	0.0166	0.0290	0.0344	0.0650	0.1004
	1.0033	1.1074	1.2521	1.4153	2.1026
	0.9214	0.9290	1.0712	1.0504	1.6463

Table D.6: Average, Standard Deviation, Maximum, and Minimum HV for the MOEA/DD

MOEA/DD HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3194	1.5873	2.0439	2.2597	3.8909
	0.0018	0.0076	0.0311	0.1109	0.2203
	1.3221	1.6032	2.1062	2.4921	4.1098
	1.3125	1.5714	1.9676	2.0848	2.9535
DTLZ2	1.2989	1.5974	2.0742	1.8867	1.8691
	0.0035	0.0032	0.0175	0.3535	0.8263
	1.3062	1.6023	2.1103	2.4718	3.4529
	1.2903	1.5883	2.0366	1.0749	0.7626
DTLZ3	1.3037	1.5943	2.1110	2.4125	3.9605
	0.0043	0.0055	0.0126	0.1298	0.1347
	1.3102	1.6048	2.1303	2.5603	4.1345
	1.2950	1.5829	2.0780	1.8253	3.6232
DTLZ4	1.2808	1.5998	2.1126	2.2087	2.2679
	0.0084	0.0048	0.0296	0.3791	0.0000
	1.2980	1.6057	2.1417	2.5819	2.2679
	1.2645	1.5844	1.9940	1.3618	2.2679
DTLZ5	1.2186	1.4270	1.5602	1.2125	1.5446
	0.0048	0.0119	0.1998	0.3187	0.3664
	1.2291	1.4469	1.8177	2.0982	3.2358
	1.2048	1.4025	1.2392	0.6325	1.0963
DTLZ6	0.9826	1.3368	1.6664	0.8263	0.6826
	0.0137	0.0136	0.0536	0.3661	0.6871
	1.0057	1.3674	1.7735	1.3521	2.1980
	0.9496	1.3115	1.5409	0.0906	0.1483

Table D.6: Average, Standard Deviation, Maximum, and Minimum HV for the MOEA/DD (continue)

MOEA/DD HV					
Benchmark Function	n_m				
	3	5	8	10	15
DTLZ7	0.4057	0.8051	0.2905	0.0254	0.0026
	0.0207	0.1002	0.2867	0.0601	0.0054
	0.4455	0.9796	1.1046	0.2421	0.0242
	0.3673	0.6142	0.0030	0.0001	0.0000
WFG1	1.0339	1.1895	0.6406	0.6131	0.0376
	0.0395	0.0531	0.0862	0.0787	0.0000
	1.0748	1.2576	0.8407	0.7933	0.0376
	0.9373	1.0182	0.5413	0.4682	0.0376
WFG2	0.9141	0.8671	0.4583	0.3075	1.6708
	0.0378	0.0735	0.0819	0.1224	0.5101
	0.9604	0.9845	0.6030	0.7763	2.6114
	0.7894	0.7269	0.2303	0.1492	0.1536
WFG3	0.6405	0.5787	0.4179	0.1418	0.2567
	0.0226	0.0232	0.0492	0.0374	0.2472
	0.6879	0.6334	0.5146	0.2608	0.9498
	0.5832	0.5472	0.3343	0.0855	0.0007
WFG4	0.6809	0.8956	0.9389	0.9928	1.9032
	0.0209	0.0436	0.0319	0.0722	0.1855
	0.7207	0.9731	1.0174	1.1048	2.2652
	0.6406	0.8096	0.8734	0.7512	1.4921
WFG5	0.7551	0.8562	0.8273	0.3670	1.2227
	0.0149	0.0289	0.0550	0.1812	0.5336
	0.7846	0.9071	0.9083	0.8451	2.1641
	0.7204	0.7985	0.6737	0.0999	0.4126

Table D.6: Average, Standard Deviation, Maximum, and Minimum HV for the MOEA/DD (continue)

MOEA/DD HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6411	0.7643	0.7227	0.4338	1.0618
	0.0208	0.0253	0.0374	0.1382	0.4212
	0.6737	0.8332	0.8033	0.6841	1.9546
	0.5927	0.7057	0.6450	0.1529	0.1752
WFG7	0.7065	0.8736	0.7958	0.6680	1.6023
	0.0129	0.0325	0.0433	0.1439	0.4159
	0.7353	0.9432	0.8748	0.9079	2.2405
	0.6755	0.8069	0.7043	0.3718	0.3584
WFG8	0.7359	0.8084	0.7670	0.3963	1.3243
	0.0147	0.0362	0.0496	0.2255	0.4425
	0.7718	0.8775	0.8799	0.8081	2.0602
	0.7113	0.6856	0.6773	0.0770	0.5855
WFG9	0.7371	0.7940	0.7599	0.6834	1.7060
	0.0433	0.0424	0.0604	0.0992	0.3945
	0.8216	0.9024	0.8804	0.8736	2.2937
	0.6735	0.6930	0.6445	0.4003	0.3990

Table D.7: Average, Standard Deviation, Maximum, and Minimum HV for the NSGA-III

NSGA-III HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3202	1.5917	2.0503	2.3253	3.6352
	0.0013	0.0042	0.0226	0.1811	0.2618
	1.3228	1.6021	2.1034	2.4830	3.9272
	1.3176	1.5823	2.0062	1.6322	3.0755
DTLZ2	1.2941	1.5856	2.0294	2.2546	3.0517
	0.0034	0.0077	0.0445	0.2915	0.4667
	1.2997	1.5949	2.1086	2.4999	3.7440
	1.2862	1.5566	1.9047	1.1584	1.9614
DTLZ3	1.3094	1.6004	2.1019	2.4613	3.9155
	0.0026	0.0038	0.0162	0.0775	0.1336
	1.3138	1.6051	2.1255	2.5745	4.1169
	1.3041	1.5910	2.0409	2.2853	3.5855
DTLZ4	1.2212	1.5498	1.9929	2.1322	1.8881
	0.0622	0.0508	0.1217	0.3854	0.1997
	1.2748	1.5929	2.1194	2.5478	2.3134
	1.0229	1.3997	1.6547	1.0575	1.7039
DTLZ5	1.2090	1.3463	1.5619	0.8290	1.3721
	0.0075	0.0914	0.2003	0.2401	0.2733
	1.2204	1.4296	1.7549	1.5281	1.8713
	1.1885	1.0795	1.0438	0.3627	0.7473
DTLZ6	0.9639	1.2864	1.5525	1.4912	1.5942
	0.0186	0.0160	0.0708	0.2774	0.5122
	1.0078	1.3090	1.6874	1.9209	2.4184
	0.9275	1.2469	1.3877	0.5769	0.2080

Table D.7: Average, Standard Deviation, Maximum, and Minimum HV for the NSGA-III (continue)

NSGA-III HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.3420	0.7503	0.8691	0.6842	0.5736
	0.0119	0.0381	0.0897	0.1550	0.1490
	0.3696	0.8425	1.0078	1.0292	0.9214
	0.3137	0.6811	0.6734	0.4115	0.2855
WFG1	0.9928	1.1988	0.6992	0.5074	0.1388
	0.0775	0.0652	0.1756	0.1645	0.2008
	1.0567	1.2593	0.9767	0.7485	0.4975
	0.7499	1.0184	0.3696	0.1519	0.0340
WFG2	0.8782	0.9296	0.6609	0.7488	2.6782
	0.0564	0.0570	0.0538	0.1120	0.3327
	0.9876	1.0350	0.8046	1.0195	3.3423
	0.8114	0.8253	0.5594	0.5714	1.7894
WFG3	0.7288	0.6836	0.5120	0.4791	1.6192
	0.0076	0.0167	0.0401	0.0653	0.2550
	0.7433	0.7170	0.5907	0.6472	2.1022
	0.7125	0.6538	0.4302	0.3925	1.0950
WFG4	0.7333	0.8303	0.9113	0.9579	1.8599
	0.0138	0.0380	0.0294	0.0863	0.1198
	0.7642	0.9327	0.9637	1.0998	2.0831
	0.7067	0.7536	0.8567	0.7602	1.6638
WFG5	0.7705	0.8728	0.8595	0.6607	1.6600
	0.0132	0.0246	0.0377	0.0792	0.1409
	0.8071	0.9238	0.9439	0.8168	1.8615
	0.7436	0.8247	0.8010	0.4925	1.3849

Table D.7: Average, Standard Deviation, Maximum, and Minimum HV for the NSGA-III (continue)

NSGA-III HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6497	0.7540	0.7447	0.5952	1.6889
	0.0128	0.0267	0.0292	0.0801	0.1538
	0.6780	0.7951	0.8379	0.7588	1.9624
	0.6238	0.6903	0.7009	0.4388	1.4184
WFG7	0.7361	0.8740	0.8493	0.8136	1.8069
	0.0158	0.0483	0.0273	0.0942	0.1558
	0.7660	0.9482	0.9096	1.0126	2.0538
	0.6867	0.7553	0.7951	0.5494	1.3214
WFG8	0.7532	0.7869	0.8290	0.6827	1.6077
	0.0152	0.0250	0.0431	0.0931	0.1140
	0.7772	0.8343	0.9180	0.8677	1.8350
	0.7179	0.7124	0.7487	0.4684	1.4221
WFG9	0.7366	0.8069	0.8691	0.7512	1.9323
	0.0256	0.0478	0.0523	0.0909	0.1445
	0.8182	0.9345	0.9873	0.9109	2.2190
	0.7039	0.7033	0.7509	0.5591	1.7005

Table D.8: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_R$

PMGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.2315	1.4765	1.8731	2.2613	3.6508
	0.0052	0.0123	0.0254	0.0365	0.0698
	1.2402	1.5024	1.9545	2.3399	3.8277
	1.2208	1.4429	1.8229	2.2052	3.5107
DTLZ2	1.2529	1.4960	1.9382	2.2479	3.4265
	0.0093	0.0213	0.0298	0.0586	0.1106
	1.2648	1.5414	1.9908	2.3595	3.6114
	1.2178	1.4636	1.8710	2.1092	3.2438
DTLZ3	1.1372	1.3767	1.7894	2.0129	3.1670
	0.0191	0.0294	0.0411	0.0902	0.2239
	1.1809	1.4429	1.8872	2.2190	3.6471
	1.1053	1.3335	1.7310	1.7975	2.7072
DTLZ4	1.2989	1.2263	1.4994	1.7609	2.5281
	0.0047	0.0296	0.0400	0.0469	0.0595
	1.3085	1.2822	1.5743	1.8449	2.6320
	1.2865	1.1606	1.4223	1.6652	2.3985
DTLZ5	1.2034	1.3623	1.7043	1.9621	2.9463
	0.0101	0.0255	0.0309	0.0454	0.0951
	1.2204	1.4010	1.7811	2.0610	3.1408
	1.1675	1.2752	1.6309	1.8515	2.7136
DTLZ6	0.7533	1.0589	1.3385	1.4256	1.9649
	0.0096	0.0150	0.0290	0.0523	0.1028
	0.7702	1.0906	1.3900	1.5153	2.1619
	0.7240	1.0287	1.2614	1.2798	1.7099

Table D.8: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_R$ (continue)

PMGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.0316	1.2025	1.1920	1.0537	1.2473
	0.0746	0.0738	0.1261	0.1555	0.2385
	1.1732	1.3049	1.3829	1.2867	1.7374
	0.8771	1.0362	0.8969	0.7488	0.6618
WFG1	0.4211	0.4830	0.5111	0.4694	0.0620
	0.0201	0.0123	0.0172	0.0201	0.0037
	0.4668	0.5114	0.5501	0.5029	0.0694
	0.3838	0.4557	0.4734	0.4242	0.0541
WFG2	0.8094	0.9148	1.0850	1.2481	2.2177
	0.0084	0.0125	0.0263	0.0534	0.0890
	0.8256	0.9385	1.1369	1.3355	2.3877
	0.7930	0.8843	1.0267	1.1150	2.0619
WFG3	0.7065	0.8100	1.0054	1.1529	2.0165
	0.0106	0.0156	0.0358	0.0410	0.0854
	0.7240	0.8393	1.0718	1.2285	2.1637
	0.6869	0.7793	0.9114	1.0618	1.8555
WFG4	0.7327	0.7607	1.0314	1.1632	1.8672
	0.0218	0.0222	0.0240	0.0304	0.0503
	0.7793	0.7968	1.0726	1.2232	1.9621
	0.6872	0.7145	0.9801	1.1157	1.7630
WFG5	0.6454	0.6824	0.7721	0.7776	1.4169
	0.0327	0.0265	0.0289	0.0345	0.0689
	0.7180	0.7251	0.8600	0.8280	1.5935
	0.5611	0.6179	0.7242	0.6791	1.3119

Table D.8: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_R$ (continue)

PMGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.5972	0.6520	0.7879	0.8827	1.4199
	0.0176	0.0213	0.0262	0.0265	0.0494
	0.6262	0.6866	0.8370	0.9264	1.5070
	0.5627	0.6107	0.7270	0.8370	1.2830
WFG7	0.5450	0.6149	0.7459	0.8150	1.2654
	0.0154	0.0162	0.0225	0.0290	0.0496
	0.5928	0.6541	0.7940	0.8922	1.3673
	0.5265	0.5921	0.6868	0.7518	1.1463
WFG8	0.5192	0.5613	0.6899	0.7686	1.2288
	0.0073	0.0172	0.0152	0.0287	0.0414
	0.5305	0.5948	0.7180	0.8157	1.2999
	0.5036	0.5218	0.6557	0.6970	1.1111
WFG9	0.8604	0.8891	1.0821	1.1928	1.9193
	0.0229	0.0219	0.0252	0.0644	0.0861
	0.9096	0.9212	1.1367	1.3086	2.0842
	0.8175	0.8410	1.0326	1.0580	1.7665

Table D.9: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{RI}$

PMGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.2313	1.4774	1.8715	2.2605	3.6439
	0.0059	0.0118	0.0263	0.0335	0.0701
	1.2449	1.5044	1.9545	2.3399	3.8277
	1.2176	1.4429	1.8172	2.2015	3.4032
DTLZ2	1.2525	1.4968	1.9403	2.2504	3.4337
	0.0109	0.0191	0.0312	0.0527	0.1161
	1.2715	1.5436	2.0029	2.3595	3.6692
	1.2178	1.4636	1.8710	2.1092	3.2438
DTLZ3	1.1376	1.3792	1.7906	2.0306	3.2072
	0.0179	0.0296	0.0484	0.0912	0.2024
	1.1809	1.4558	1.9326	2.2674	3.6730
	1.1010	1.3256	1.7072	1.7975	2.7072
DTLZ4	1.2998	1.2307	1.4849	1.7599	2.5231
	0.0043	0.0279	0.0493	0.0426	0.0564
	1.3085	1.3188	1.5743	1.8449	2.6320
	1.2865	1.1606	1.3343	1.6652	2.3978
DTLZ5	1.2024	1.3662	1.7045	1.9677	2.9524
	0.0095	0.0219	0.0334	0.0526	0.0879
	1.2204	1.4100	1.7811	2.0622	3.1408
	1.1675	1.2752	1.6309	1.8412	2.7136
DTLZ6	0.7544	1.0598	1.3406	1.4125	1.9575
	0.0085	0.0134	0.0288	0.0523	0.1023
	0.7702	1.0906	1.4022	1.5153	2.1619
	0.7240	1.0287	1.2614	1.2798	1.7099

Table D.9: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{RI}$ (continue)

PMGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.0331	1.2087	1.1883	1.0180	1.2069
	0.0771	0.0858	0.1284	0.1621	0.2297
	1.2336	1.3523	1.4522	1.2867	1.7374
	0.8771	0.9320	0.8379	0.6361	0.6618
WFG1	0.4210	0.4792	0.5102	0.4691	0.0626
	0.0183	0.0168	0.0216	0.0189	0.0042
	0.4668	0.5190	0.5734	0.5116	0.0735
	0.3838	0.4378	0.4686	0.4242	0.0541
WFG2	0.8086	0.9153	1.0856	1.2465	2.2069
	0.0084	0.0139	0.0281	0.0551	0.0910
	0.8288	0.9554	1.1484	1.3501	2.3907
	0.7914	0.8843	1.0267	1.1057	1.9744
WFG3	0.7070	0.8108	1.0124	1.1542	2.0034
	0.0106	0.0147	0.0314	0.0406	0.0963
	0.7283	0.8393	1.0718	1.2285	2.1637
	0.6869	0.7793	0.9114	1.0618	1.6978
WFG4	0.7308	0.7649	1.0271	1.1555	1.8697
	0.0223	0.0228	0.0243	0.0330	0.0498
	0.7793	0.8126	1.0776	1.2560	1.9621
	0.6767	0.7145	0.9721	1.0979	1.7630
WFG5	0.6416	0.6852	0.7725	0.7861	1.4132
	0.0305	0.0260	0.0288	0.0363	0.0683
	0.7180	0.7399	0.8600	0.8602	1.5935
	0.5611	0.6179	0.7184	0.6791	1.2236

Table D.9: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{RI}$ (continue)

PMGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.5970	0.6529	0.7908	0.8822	1.4271
	0.0195	0.0189	0.0256	0.0272	0.0484
	0.6333	0.6866	0.8407	0.9264	1.5207
	0.5627	0.5992	0.7270	0.8158	1.2830
WFG7	0.5441	0.6147	0.7477	0.8146	1.2613
	0.0145	0.0167	0.0248	0.0280	0.0470
	0.5928	0.6541	0.8040	0.8922	1.3673
	0.5184	0.5895	0.6868	0.7518	1.1463
WFG8	0.5192	0.5586	0.6888	0.7699	1.2349
	0.0094	0.0159	0.0162	0.0288	0.0485
	0.5420	0.5948	0.7261	0.8301	1.3521
	0.4957	0.5171	0.6557	0.6860	1.1111
WFG9	0.8604	0.8875	1.0867	1.1948	1.9184
	0.0251	0.0228	0.0257	0.0560	0.0944
	0.9322	0.9327	1.1510	1.3086	2.0842
	0.8062	0.8410	1.0326	1.0580	1.5876

Table D.10: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{STD}$

PMGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.2378	1.4763	1.8769	2.2593	3.6442
	0.0186	0.0115	0.0377	0.0396	0.0634
	1.3201	1.5044	2.0855	2.3964	3.8277
	1.2176	1.4429	1.8172	2.1796	3.4032
DTLZ2	1.2569	1.5001	1.9404	2.2532	3.4378
	0.0129	0.0198	0.0333	0.0562	0.1162
	1.2871	1.5513	2.0029	2.3719	3.7135
	1.2178	1.4634	1.8590	2.1092	3.2317
DTLZ3	1.1530	1.3856	1.7908	2.0227	3.2282
	0.0325	0.0311	0.0508	0.0859	0.1994
	1.2553	1.4558	1.9326	2.2674	3.6730
	1.1003	1.3256	1.6321	1.7975	2.7072
DTLZ4	1.2981	1.2405	1.5019	1.7796	2.5498
	0.0075	0.0302	0.0495	0.0497	0.0652
	1.3085	1.3229	1.5955	1.8946	2.6770
	1.2505	1.1606	1.3343	1.6652	2.3978
DTLZ5	1.2078	1.3711	1.7140	1.9767	2.9512
	0.0139	0.0268	0.0390	0.0588	0.0937
	1.2441	1.4422	1.8028	2.1029	3.1408
	1.1675	1.2752	1.6209	1.8279	2.6384
DTLZ6	0.7531	1.0587	1.3434	1.4108	1.9716
	0.0091	0.0131	0.0290	0.0518	0.1023
	0.7702	1.0906	1.4022	1.5288	2.2710
	0.7240	1.0287	1.2614	1.2798	1.7099

Table D.10: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{STD}$ (continue)

PMGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.0247	1.2112	1.2240	1.1256	1.4608
	0.0759	0.0948	0.1466	0.2519	0.4522
	1.2336	1.3975	1.5657	1.7031	2.4167
	0.8682	0.9320	0.8379	0.6361	0.6618
WFG1	0.4250	0.4806	0.5098	0.4637	0.0628
	0.0220	0.0217	0.0242	0.0229	0.0042
	0.4885	0.5569	0.5734	0.5152	0.0735
	0.3638	0.4265	0.4535	0.4047	0.0541
WFG2	0.8078	0.9185	1.0940	1.2539	2.1999
	0.0094	0.0142	0.0302	0.0544	0.0953
	0.8288	0.9554	1.1671	1.3501	2.3907
	0.7792	0.8843	1.0267	1.1057	1.9744
WFG3	0.7083	0.8166	1.0206	1.1622	1.9916
	0.0127	0.0170	0.0395	0.0482	0.0931
	0.7405	0.8594	1.0936	1.2943	2.1705
	0.6765	0.7793	0.9114	1.0618	1.6978
WFG4	0.7272	0.7646	1.0241	1.1497	1.8591
	0.0227	0.0234	0.0253	0.0340	0.0538
	0.7793	0.8126	1.0776	1.2560	1.9621
	0.6668	0.7145	0.9721	1.0722	1.7310
WFG5	0.6423	0.6877	0.7831	0.8046	1.4449
	0.0301	0.0267	0.0380	0.0507	0.0950
	0.7192	0.7573	0.9013	0.9676	1.7255
	0.5611	0.6179	0.7116	0.6791	1.2236

Table D.10: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{STD}$ (continue)

PMGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.5948	0.6497	0.7905	0.8793	1.4194
	0.0189	0.0221	0.0290	0.0332	0.0510
	0.6333	0.6915	0.8516	0.9533	1.5262
	0.5627	0.5961	0.7270	0.8011	1.2830
WFG7	0.5574	0.6205	0.7504	0.8208	1.2602
	0.0264	0.0203	0.0283	0.0365	0.0506
	0.6208	0.6722	0.8143	0.9228	1.3673
	0.5137	0.5895	0.6714	0.7518	1.1122
WFG8	0.5198	0.5600	0.6926	0.7872	1.2409
	0.0121	0.0155	0.0208	0.0374	0.0502
	0.5509	0.5948	0.7515	0.8758	1.3521
	0.4855	0.5171	0.6330	0.6860	1.1111
WFG9	0.8578	0.8871	1.0880	1.1961	1.9177
	0.0262	0.0237	0.0265	0.0529	0.0880
	0.9322	0.9351	1.1643	1.3086	2.0842
	0.8050	0.8410	1.0326	1.0580	1.5876

D.2 Inverted Generational Distance Values

The average, standard deviation, maximum, and minimum IGD performance measure values for each algorithm on each problem instance are listed in tables D.11 to D.20. Note that these performance measure values are associated with Chapter 4.

Table D.11: Average, Standard Deviation, Maximum, and Minimum IGD for the CDAS-SMPSO algorithm

CDAS-SMPSO IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0072	0.0068	0.0078	0.0078	0.0071
	0.0008	0.0006	0.0012	0.0025	0.0003
	0.0079	0.0076	0.0131	0.0156	0.0079
	0.0049	0.0055	0.0066	0.0060	0.0068
DTLZ2	0.0185	0.0184	0.0207	0.0187	0.0210
	0.0003	0.0005	0.0010	0.0006	0.0008
	0.0192	0.0195	0.0235	0.0202	0.0230
	0.0179	0.0175	0.0195	0.0175	0.0194
DTLZ3	0.0225	0.0258	0.0289	0.0248	0.0266
	0.0017	0.0006	0.0007	0.0011	0.0005
	0.0252	0.0271	0.0294	0.0261	0.0269
	0.0185	0.0237	0.0254	0.0228	0.0250
DTLZ4	0.0198	0.0225	0.0275	0.0254	0.0277
	0.0021	0.0013	0.0013	0.0012	0.0010
	0.0238	0.0257	0.0307	0.0280	0.0301
	0.0149	0.0205	0.0254	0.0225	0.0258

Table D.11: Average, Standard Deviation, Maximum, and Minimum IGD for the CDAS-SMPSO algorithm (continue)

CDAS-SMPSO IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ5	0.0209	0.0198	0.0187	0.0182	0.0174
	0.0002	0.0002	0.0002	0.0002	0.0002
	0.0212	0.0200	0.0190	0.0186	0.0180
	0.0204	0.0195	0.0181	0.0177	0.0168
DTLZ6	0.0231	0.0232	0.0230	0.0238	0.0239
	0.0010	0.0010	0.0015	0.0009	0.0011
	0.0251	0.0254	0.0252	0.0255	0.0262
	0.0216	0.0217	0.0183	0.0223	0.0212
DTLZ7	0.1041	0.1441	0.2399	0.0984	0.1444
	0.0012	0.0018	0.0013	0.0004	0.0005
	0.1067	0.1481	0.2428	0.0993	0.1451
	0.1025	0.1408	0.2363	0.0975	0.1433
WFG1	0.0388	0.0499	0.0818	0.0808	0.1324
	0.0030	0.0019	0.0020	0.0010	0.0002
	0.0462	0.0562	0.0860	0.0822	0.1326
	0.0360	0.0484	0.0786	0.0783	0.1321
WFG2	0.0460	0.0578	0.0925	0.0882	0.1369
	0.0011	0.0010	0.0013	0.0010	0.0008
	0.0488	0.0597	0.0957	0.0910	0.1383
	0.0442	0.0558	0.0907	0.0861	0.1355
WFG3	0.0766	0.1406	0.2429	0.3121	0.4850
	0.0016	0.0007	0.0004	0.0015	0.0010
	0.0801	0.1431	0.2439	0.3168	0.4870
	0.0736	0.1396	0.2420	0.3109	0.4837

Table D.11: Average, Standard Deviation, Maximum, and Minimum IGD for the CDAS-SMPSO algorithm (continue)

CDAS-SMPSO IGD					
Benchmark Function	n_m				
	3	5	8	10	15
WFG4	0.0881	0.1542	0.2732	0.3042	0.4692
	0.0001	0.0002	0.0002	0.0003	0.0007
	0.0883	0.1546	0.2736	0.3051	0.4706
	0.0879	0.1537	0.2728	0.3036	0.4680
WFG5	0.0888	0.1542	0.2741	0.3047	0.4708
	0.0002	0.0001	0.0002	0.0003	0.0004
	0.0893	0.1544	0.2747	0.3053	0.4717
	0.0884	0.1540	0.2737	0.3041	0.4703
WFG6	0.0880	0.1538	0.2736	0.3045	0.4697
	0.0002	0.0002	0.0003	0.0004	0.0005
	0.0884	0.1541	0.2742	0.3054	0.4714
	0.0874	0.1534	0.2732	0.3039	0.4689
WFG7	0.0885	0.1544	0.2742	0.3039	0.4686
	0.0001	0.0002	0.0002	0.0002	0.0005
	0.0886	0.1548	0.2747	0.3045	0.4698
	0.0883	0.1541	0.2737	0.3036	0.4673
WFG8	0.0889	0.1547	0.2735	0.3037	0.4692
	0.0001	0.0004	0.0001	0.0002	0.0003
	0.0893	0.1564	0.2737	0.3043	0.4700
	0.0886	0.1540	0.2731	0.3034	0.4689
WFG9	0.0900	0.1542	0.2737	0.3052	0.4694
	0.0002	0.0002	0.0006	0.0023	0.0008
	0.0906	0.1546	0.2762	0.3136	0.4720
	0.0894	0.1538	0.2730	0.3035	0.4683

Table D.12: Average, Standard Deviation, Maximum, and Minimum IGD for the KnEA

KnEA IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0024	-	0.0065	0.0057	0.0070
	0.0001	-	0.0005	0.0005	0.0003
	0.0027	-	0.0075	0.0066	0.0077
	0.0021	-	0.0055	0.0049	0.0064
DTLZ2	0.0162	0.0167	0.0267	0.0195	0.0270
	0.0004	0.0018	0.0012	0.0026	0.0010
	0.0172	0.0208	0.0291	0.0260	0.0281
	0.0151	0.0143	0.0237	0.0156	0.0243
DTLZ3	0.0168	0.0218	0.0261	0.0230	0.0254
	0.0005	0.0013	0.0013	0.0016	0.0007
	0.0183	0.0248	0.0286	0.0254	0.0267
	0.0158	0.0192	0.0223	0.0197	0.0233
DTLZ4	0.0146	0.0230	0.0291	0.0261	0.0281
	0.0015	0.0019	0.0001	0.0001	0.0002
	0.0189	0.0250	0.0294	0.0263	0.0284
	0.0130	0.0191	0.0290	0.0259	0.0271
DTLZ5	0.0175	0.0207	0.0202	0.0201	0.0188
	0.0005	0.0013	0.0004	0.0012	0.0021
	0.0181	0.0214	0.0209	0.0208	0.0215
	0.0166	0.0149	0.0190	0.0147	0.0128
DTLZ6	0.0057	0.0225	0.0222	0.0222	0.0216
	0.0004	0.0041	0.0035	0.0029	0.0033
	0.0063	0.0257	0.0256	0.0270	0.0253
	0.0050	0.0128	0.0159	0.0157	0.0153

Table D.12: Average, Standard Deviation, Maximum, and Minimum IGD for the KnEA (continue)

KnEA IGD					
Benchmark	n_m				
Function	**3**	**5**	**8**	**10**	**15**
DTLZ7	0.0996	0.1338	0.2249	0.0933	0.1384
	0.0003	0.0013	0.0013	0.0006	0.0004
	0.1002	0.1360	0.2268	0.0943	0.1395
	0.0991	0.1307	0.2215	0.0920	0.1374
WFG1	0.0471	0.0591	0.0861	0.0831	0.1327
	0.0026	0.0014	0.0022	0.0013	0.0011
	0.0489	0.0611	0.0891	0.0853	0.1339
	0.0358	0.0562	0.0814	0.0805	0.1299
WFG2	0.0450	0.0610	0.0895	0.0852	0.1356
	0.0013	0.0009	0.0010	0.0008	0.0010
	0.0472	0.0629	0.0923	0.0872	0.1382
	0.0412	0.0587	0.0880	0.0839	0.1335
WFG3	0.0783	0.1480	0.2486	0.3178	0.4878
	0.0017	0.0014	0.0024	0.0030	0.0013
	0.0818	0.1510	0.2532	0.3244	0.4907
	0.0755	0.1457	0.2435	0.3125	0.4851
WFG4	0.0882	0.1546	0.2759	0.3048	0.4731
	0.0001	0.0002	0.0015	0.0007	0.0015
	0.0884	0.1551	0.2804	0.3067	0.4770
	0.0881	0.1543	0.2740	0.3038	0.4702
WFG5	0.0891	0.1550	0.2811	0.3071	0.4721
	0.0001	0.0003	0.0034	0.0019	0.0016
	0.0893	0.1563	0.2876	0.3109	0.4769
	0.0888	0.1547	0.2755	0.3041	0.4698

Table D.12: Average, Standard Deviation, Maximum, and Minimum IGD for the KnEA (continue)

KnEA IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0881	0.1546	0.2814	0.3084	0.4736
	0.0001	0.0013	0.0028	0.0015	0.0029
	0.0884	0.1606	0.2871	0.3119	0.4822
	0.0878	0.1539	0.2756	0.3046	0.4701
WFG7	0.0884	0.1548	0.2801	0.3066	0.4703
	0.0001	0.0002	0.0018	0.0021	0.0011
	0.0887	0.1558	0.2839	0.3114	0.4729
	0.0882	0.1544	0.2766	0.3036	0.4682
WFG8	0.0892	0.1550	0.2798	0.3075	0.4712
	0.0001	0.0008	0.0021	0.0024	0.0011
	0.0894	0.1570	0.2849	0.3123	0.4737
	0.0890	0.1543	0.2763	0.3032	0.4694
WFG9	0.0894	0.1545	0.2768	0.3050	0.4696
	0.0003	0.0004	0.0017	0.0013	0.0015
	0.0901	0.1556	0.2819	0.3086	0.4741
	0.0888	0.1539	0.2738	0.3032	0.4675

Table D.13: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_R$

MGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0011	0.0053	0.0095	0.0088	0.0095
	0.0001	0.0003	0.0005	0.0006	0.0005
	0.0014	0.0058	0.0104	0.0097	0.0105
	0.0009	0.0048	0.0085	0.0072	0.0087
DTLZ2	0.0188	0.0153	0.0167	0.0156	0.0202
	0.0003	0.0004	0.0005	0.0005	0.0006
	0.0193	0.0161	0.0176	0.0165	0.0212
	0.0181	0.0143	0.0158	0.0147	0.0194
DTLZ3	0.0113	0.0103	0.0150	0.0158	0.0220
	0.0007	0.0003	0.0006	0.0006	0.0007
	0.0136	0.0108	0.0160	0.0168	0.0231
	0.0104	0.0097	0.0140	0.0147	0.0206
DTLZ4	0.0221	0.0234	0.0288	0.0260	0.0280
	0.0019	0.0010	0.0005	0.0001	0.0002
	0.0238	0.0257	0.0295	0.0265	0.0285
	0.0190	0.0220	0.0268	0.0258	0.0277
DTLZ5	0.0211	0.0165	0.0142	0.0137	0.0127
	0.0003	0.0005	0.0004	0.0004	0.0005
	0.0214	0.0172	0.0148	0.0144	0.0135
	0.0204	0.0152	0.0132	0.0130	0.0117
DTLZ6	0.0018	0.0039	0.0053	0.0065	0.0072
	0.0002	0.0005	0.0005	0.0007	0.0009
	0.0022	0.0050	0.0062	0.0081	0.0098
	0.0015	0.0029	0.0046	0.0052	0.0056

Table D.13: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_R$ (continue)

MGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1184	0.1411	0.2255	0.0925	0.1377
	0.0013	0.0038	0.0005	0.0002	0.0002
	0.1204	0.1474	0.2264	0.0929	0.1383
	0.1159	0.1351	0.2247	0.0922	0.1373
WFG1	0.0502	0.0614	0.0881	0.0842	0.1336
	0.0004	0.0005	0.0004	0.0003	0.0002
	0.0513	0.0640	0.0899	0.0856	0.1342
	0.0496	0.0610	0.0877	0.0838	0.1335
WFG2	0.0497	0.0614	0.0903	0.0860	0.1359
	0.0006	0.0005	0.0003	0.0003	0.0002
	0.0513	0.0625	0.0913	0.0866	0.1362
	0.0487	0.0607	0.0899	0.0855	0.1356
WFG3	0.0750	0.1410	0.2435	0.3120	0.4845
	0.0001	0.0001	0.0001	0.0001	0.0001
	0.0752	0.1411	0.2437	0.3121	0.4846
	0.0748	0.1408	0.2433	0.3118	0.4844
WFG4	0.0890	0.1561	0.2783	0.3089	0.4772
	0.0000	0.0001	0.0002	0.0003	0.0003
	0.0891	0.1564	0.2787	0.3093	0.4779
	0.0889	0.1560	0.2779	0.3084	0.4765
WFG5	0.0886	0.1541	0.2739	0.3040	0.4701
	0.0004	0.0002	0.0002	0.0001	0.0003
	0.0893	0.1545	0.2742	0.3043	0.4707
	0.0878	0.1538	0.2736	0.3037	0.4695

Table D.13: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_R$ (continue)

MGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0909	0.1566	0.2760	0.3059	0.4731
	0.0002	0.0002	0.0002	0.0002	0.0002
	0.0911	0.1569	0.2764	0.3063	0.4734
	0.0905	0.1563	0.2757	0.3055	0.4726
WFG7	0.0902	0.1572	0.2779	0.3074	0.4729
	0.0001	0.0001	0.0002	0.0002	0.0003
	0.0903	0.1574	0.2785	0.3078	0.4735
	0.0900	0.1570	0.2774	0.3070	0.4725
WFG8	0.0906	0.1573	0.2771	0.3066	0.4737
	0.0001	0.0002	0.0002	0.0002	0.0003
	0.0909	0.1577	0.2774	0.3070	0.4745
	0.0904	0.1569	0.2766	0.3063	0.4731
WFG9	0.0910	0.1552	0.2735	0.3034	0.4676
	0.0002	0.0003	0.0001	0.0002	0.0002
	0.0914	0.1558	0.2738	0.3038	0.4680
	0.0903	0.1547	0.2732	0.3031	0.4672

Table D.14: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{RI}$

MGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0011	0.0053	0.0095	0.0089	0.0096
	0.0001	0.0003	0.0005	0.0006	0.0006
	0.0014	0.0059	0.0104	0.0103	0.0110
	0.0009	0.0048	0.0084	0.0072	0.0082
DTLZ2	0.0188	0.0154	0.0167	0.0156	0.0202
	0.0003	0.0004	0.0004	0.0004	0.0005
	0.0193	0.0165	0.0176	0.0165	0.0212
	0.0181	0.0143	0.0157	0.0147	0.0194
DTLZ3	0.0112	0.0102	0.0150	0.0159	0.0220
	0.0007	0.0003	0.0005	0.0006	0.0006
	0.0136	0.0109	0.0163	0.0173	0.0231
	0.0098	0.0095	0.0140	0.0147	0.0206
DTLZ4	0.0221	0.0235	0.0288	0.0260	0.0280
	0.0018	0.0010	0.0006	0.0001	0.0002
	0.0238	0.0257	0.0301	0.0265	0.0285
	0.0190	0.0220	0.0264	0.0258	0.0277
DTLZ5	0.0211	0.0165	0.0142	0.0137	0.0127
	0.0003	0.0005	0.0004	0.0004	0.0005
	0.0214	0.0174	0.0149	0.0146	0.0136
	0.0201	0.0152	0.0132	0.0127	0.0117
DTLZ6	0.0018	0.0038	0.0053	0.0064	0.0071
	0.0002	0.0005	0.0005	0.0007	0.0008
	0.0023	0.0050	0.0064	0.0082	0.0098
	0.0014	0.0028	0.0043	0.0048	0.0056

Table D.14: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{RI}$ (continue)

MGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1182	0.1416	0.2255	0.0925	0.1378
	0.0013	0.0038	0.0004	0.0002	0.0003
	0.1204	0.1474	0.2264	0.0930	0.1384
	0.1151	0.1351	0.2247	0.0922	0.1370
WFG1	0.0502	0.0615	0.0881	0.0842	0.1336
	0.0004	0.0008	0.0005	0.0004	0.0001
	0.0513	0.0645	0.0903	0.0859	0.1342
	0.0496	0.0610	0.0877	0.0838	0.1335
WFG2	0.0498	0.0614	0.0902	0.0860	0.1359
	0.0006	0.0005	0.0003	0.0002	0.0002
	0.0513	0.0628	0.0913	0.0866	0.1363
	0.0487	0.0607	0.0897	0.0855	0.1355
WFG3	0.0750	0.1410	0.2435	0.3120	0.4845
	0.0001	0.0001	0.0001	0.0001	0.0001
	0.0752	0.1411	0.2437	0.3122	0.4847
	0.0748	0.1408	0.2433	0.3118	0.4844
WFG4	0.0890	0.1561	0.2782	0.3088	0.4771
	0.0000	0.0001	0.0002	0.0002	0.0003
	0.0891	0.1565	0.2787	0.3093	0.4779
	0.0889	0.1559	0.2779	0.3084	0.4765
WFG5	0.0886	0.1541	0.2738	0.3040	0.4700
	0.0004	0.0002	0.0002	0.0002	0.0003
	0.0893	0.1545	0.2743	0.3044	0.4707
	0.0878	0.1536	0.2736	0.3036	0.4695

Table D.14: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{RI}$ (continue)

MGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0909	0.1566	0.2760	0.3059	0.4731
	0.0001	0.0002	0.0002	0.0002	0.0002
	0.0911	0.1569	0.2764	0.3063	0.4735
	0.0905	0.1561	0.2757	0.3055	0.4726
WFG7	0.0902	0.1572	0.2779	0.3075	0.4729
	0.0001	0.0001	0.0002	0.0002	0.0003
	0.0904	0.1574	0.2785	0.3079	0.4735
	0.0899	0.1570	0.2774	0.3070	0.4722
WFG8	0.0906	0.1573	0.2771	0.3066	0.4737
	0.0001	0.0002	0.0002	0.0002	0.0003
	0.0909	0.1577	0.2776	0.3070	0.4745
	0.0903	0.1568	0.2766	0.3063	0.4731
WFG9	0.0910	0.1551	0.2735	0.3034	0.4676
	0.0002	0.0003	0.0001	0.0001	0.0002
	0.0914	0.1558	0.2740	0.3038	0.4680
	0.0903	0.1547	0.2732	0.3031	0.4671

Table D.15: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{STD}$

MGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0018	0.0052	0.0094	0.0089	0.0095
	0.0013	0.0003	0.0005	0.0007	0.0006
	0.0070	0.0059	0.0105	0.0103	0.0110
	0.0009	0.0043	0.0077	0.0059	0.0082
DTLZ2	0.0187	0.0153	0.0167	0.0156	0.0202
	0.0004	0.0004	0.0004	0.0004	0.0005
	0.0193	0.0165	0.0179	0.0165	0.0212
	0.0175	0.0143	0.0157	0.0146	0.0194
DTLZ3	0.0123	0.0104	0.0152	0.0159	0.0220
	0.0021	0.0005	0.0014	0.0006	0.0006
	0.0202	0.0133	0.0271	0.0177	0.0231
	0.0098	0.0095	0.0140	0.0147	0.0206
DTLZ4	0.0221	0.0236	0.0286	0.0260	0.0279
	0.0018	0.0011	0.0006	0.0001	0.0002
	0.0238	0.0257	0.0301	0.0265	0.0285
	0.0190	0.0220	0.0264	0.0257	0.0276
DTLZ5	0.0210	0.0164	0.0142	0.0137	0.0127
	0.0004	0.0005	0.0004	0.0004	0.0005
	0.0214	0.0174	0.0150	0.0146	0.0137
	0.0196	0.0152	0.0132	0.0124	0.0117
DTLZ6	0.0018	0.0039	0.0052	0.0064	0.0071
	0.0002	0.0005	0.0005	0.0007	0.0009
	0.0023	0.0057	0.0064	0.0082	0.0098
	0.0014	0.0028	0.0039	0.0048	0.0051

Table D.15: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{STD}$ (continue)

MGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1176	0.1419	0.2258	0.0926	0.1380
	0.0015	0.0036	0.0008	0.0003	0.0005
	0.1204	0.1474	0.2284	0.0934	0.1401
	0.1142	0.1334	0.2247	0.0922	0.1370
WFG1	0.0499	0.0614	0.0880	0.0843	0.1337
	0.0005	0.0007	0.0004	0.0003	0.0002
	0.0513	0.0645	0.0903	0.0859	0.1351
	0.0492	0.0608	0.0876	0.0838	0.1335
WFG2	0.0498	0.0614	0.0902	0.0860	0.1359
	0.0006	0.0005	0.0003	0.0003	0.0002
	0.0513	0.0628	0.0913	0.0873	0.1363
	0.0487	0.0607	0.0897	0.0855	0.1351
WFG3	0.0750	0.1409	0.2435	0.3120	0.4845
	0.0001	0.0001	0.0001	0.0001	0.0001
	0.0752	0.1412	0.2437	0.3122	0.4847
	0.0747	0.1407	0.2433	0.3118	0.4843
WFG4	0.0890	0.1561	0.2782	0.3088	0.4771
	0.0000	0.0001	0.0002	0.0002	0.0003
	0.0892	0.1565	0.2788	0.3093	0.4779
	0.0889	0.1559	0.2778	0.3083	0.4763
WFG5	0.0885	0.1541	0.2738	0.3040	0.4700
	0.0005	0.0002	0.0002	0.0002	0.0003
	0.0900	0.1545	0.2743	0.3044	0.4707
	0.0876	0.1536	0.2736	0.3036	0.4695

Table D.15: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{STD}$ (continue)

MGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0908	0.1565	0.2760	0.3059	0.4731
	0.0002	0.0002	0.0002	0.0002	0.0002
	0.0911	0.1569	0.2764	0.3063	0.4735
	0.0901	0.1561	0.2757	0.3055	0.4726
WFG7	0.0901	0.1572	0.2778	0.3074	0.4729
	0.0001	0.0001	0.0002	0.0002	0.0003
	0.0904	0.1575	0.2785	0.3079	0.4737
	0.0897	0.1569	0.2773	0.3069	0.4722
WFG8	0.0906	0.1573	0.2771	0.3066	0.4737
	0.0001	0.0002	0.0002	0.0002	0.0003
	0.0909	0.1578	0.2776	0.3071	0.4746
	0.0903	0.1568	0.2766	0.3063	0.4731
WFG9	0.0910	0.1551	0.2735	0.3034	0.4676
	0.0002	0.0003	0.0001	0.0001	0.0002
	0.0914	0.1559	0.2740	0.3038	0.4681
	0.0903	0.1546	0.2732	0.3031	0.4671

Table D.16: Average, Standard Deviation, Maximum, and Minimum IGD for the MOEA/DD

MOEA/DD IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0012	0.0025	0.0054	0.0054	0.0067
	0.0001	0.0001	0.0003	0.0004	0.0008
	0.0015	0.0029	0.0065	0.0067	0.0103
	0.0010	0.0023	0.0047	0.0047	0.0059
DTLZ2	0.0125	0.0128	0.0168	0.0202	0.0262
	0.0006	0.0004	0.0008	0.0021	0.0022
	0.0137	0.0137	0.0184	0.0247	0.0321
	0.0114	0.0118	0.0150	0.0162	0.0232
DTLZ3	0.0124	0.0144	0.0187	0.0204	0.0235
	0.0004	0.0005	0.0007	0.0009	0.0008
	0.0131	0.0157	0.0201	0.0219	0.0257
	0.0116	0.0136	0.0171	0.0186	0.0217
DTLZ4	0.0101	0.0134	0.0196	0.0224	0.0286
	0.0011	0.0018	0.0018	0.0015	0.0000
	0.0122	0.0170	0.0230	0.0255	0.0286
	0.0079	0.0111	0.0169	0.0194	0.0286
DTLZ5	0.0125	0.0132	0.0128	0.0153	0.0155
	0.0006	0.0006	0.0006	0.0022	0.0006
	0.0138	0.0143	0.0140	0.0207	0.0167
	0.0112	0.0121	0.0115	0.0129	0.0146
DTLZ6	0.0038	0.0069	0.0088	0.0190	0.0273
	0.0003	0.0005	0.0007	0.0052	0.0055
	0.0045	0.0079	0.0101	0.0314	0.0332
	0.0032	0.0058	0.0072	0.0128	0.0153

Table D.16: Average, Standard Deviation, Maximum, and Minimum IGD for the MOEA/DD (continue)

MOEA/DD IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1002	0.1401	0.2353	0.0980	0.1435
	0.0004	0.0005	0.0008	0.0003	0.0002
	0.1009	0.1409	0.2371	0.0985	0.1442
	0.0995	0.1391	0.2341	0.0972	0.1430
WFG1	0.0389	0.0538	0.0882	0.0843	0.1343
	0.0033	0.0029	0.0005	0.0002	0.0000
	0.0451	0.0577	0.0890	0.0848	0.1343
	0.0349	0.0483	0.0870	0.0838	0.1343
WFG2	0.0442	0.0618	0.0907	0.0881	0.1351
	0.0009	0.0012	0.0021	0.0015	0.0011
	0.0458	0.0649	0.0960	0.0903	0.1370
	0.0417	0.0604	0.0879	0.0849	0.1327
WFG3	0.0770	0.1463	0.2467	0.3186	0.4894
	0.0011	0.0013	0.0028	0.0014	0.0027
	0.0787	0.1486	0.2498	0.3224	0.4990
	0.0743	0.1439	0.2426	0.3130	0.4843
WFG4	0.0883	0.1544	0.2744	0.3051	0.4739
	0.0004	0.0002	0.0004	0.0006	0.0015
	0.0893	0.1551	0.2750	0.3063	0.4782
	0.0877	0.1541	0.2736	0.3042	0.4715
WFG5	0.0886	0.1555	0.2744	0.3103	0.4736
	0.0001	0.0007	0.0005	0.0037	0.0017
	0.0889	0.1572	0.2761	0.3198	0.4771
	0.0884	0.1545	0.2738	0.3041	0.4707

Table D.16: Average, Standard Deviation, Maximum, and Minimum IGD for the MOEA/DD (continue)

MOEA/DD IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0875	0.1543	0.2737	0.3080	0.4733
	0.0002	0.0006	0.0004	0.0029	0.0020
	0.0878	0.1556	0.2751	0.3156	0.4787
	0.0871	0.1536	0.2732	0.3044	0.4703
WFG7	0.0880	0.1552	0.2740	0.3053	0.4714
	0.0002	0.0007	0.0002	0.0012	0.0013
	0.0888	0.1565	0.2744	0.3094	0.4760
	0.0876	0.1542	0.2735	0.3036	0.4698
WFG8	0.0888	0.1543	0.2736	0.3062	0.4721
	0.0002	0.0002	0.0003	0.0028	0.0016
	0.0892	0.1550	0.2742	0.3123	0.4761
	0.0884	0.1539	0.2732	0.3031	0.4695
WFG9	0.0881	0.1538	0.2732	0.3040	0.4707
	0.0004	0.0002	0.0002	0.0007	0.0009
	0.0888	0.1543	0.2737	0.3067	0.4729
	0.0874	0.1534	0.2728	0.3031	0.4679

Table D.17: Average, Standard Deviation, Maximum, and Minimum IGD for the NSGA-III

NSGA-III IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0013	0.0031	0.0074	0.0069	0.0087
	0.0001	0.0003	0.0006	0.0009	0.0010
	0.0016	0.0041	0.0089	0.0100	0.0109
	0.0011	0.0027	0.0065	0.0056	0.0071
DTLZ2	0.0138	0.0111	0.0162	0.0173	0.0216
	0.0004	0.0004	0.0022	0.0014	0.0011
	0.0145	0.0120	0.0221	0.0206	0.0238
	0.0126	0.0106	0.0138	0.0148	0.0198
DTLZ3	0.0144	0.0132	0.0176	0.0178	0.0203
	0.0004	0.0005	0.0017	0.0011	0.0007
	0.0152	0.0140	0.0203	0.0198	0.0223
	0.0138	0.0122	0.0148	0.0159	0.0191
DTLZ4	0.0132	0.0140	0.0206	0.0229	0.0297
	0.0029	0.0035	0.0027	0.0021	0.0009
	0.0177	0.0217	0.0262	0.0267	0.0310
	0.0104	0.0096	0.0156	0.0193	0.0282
DTLZ5	0.0151	0.0126	0.0110	0.0163	0.0140
	0.0005	0.0009	0.0012	0.0027	0.0025
	0.0161	0.0150	0.0136	0.0242	0.0230
	0.0141	0.0109	0.0089	0.0128	0.0112
DTLZ6	0.0041	0.0073	0.0099	0.0152	0.0176
	0.0004	0.0006	0.0019	0.0044	0.0050
	0.0052	0.0081	0.0145	0.0230	0.0284
	0.0035	0.0060	0.0070	0.0079	0.0085

Table D.17: Average, Standard Deviation, Maximum, and Minimum IGD for the NSGA-III (continue)

NSGA-III IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.0980	0.1318	0.2221	0.0930	0.1385
	0.0004	0.0010	0.0008	0.0003	0.0004
	0.0987	0.1338	0.2233	0.0937	0.1394
	0.0969	0.1300	0.2204	0.0925	0.1379
WFG1	0.0394	0.0514	0.0836	0.0814	0.1328
	0.0035	0.0026	0.0028	0.0020	0.0012
	0.0474	0.0563	0.0880	0.0833	0.1341
	0.0350	0.0476	0.0758	0.0765	0.1309
WFG2	0.0439	0.0589	0.0877	0.0843	0.1332
	0.0009	0.0013	0.0010	0.0009	0.0006
	0.0460	0.0606	0.0893	0.0861	0.1343
	0.0421	0.0554	0.0854	0.0823	0.1316
WFG3	0.0736	0.1400	0.2451	0.3169	0.4882
	0.0007	0.0004	0.0012	0.0015	0.0018
	0.0752	0.1410	0.2484	0.3205	0.4911
	0.0726	0.1393	0.2431	0.3131	0.4854
WFG4	0.0878	0.1542	0.2743	0.3049	0.4725
	0.0001	0.0002	0.0002	0.0002	0.0005
	0.0880	0.1550	0.2748	0.3054	0.4737
	0.0877	0.1539	0.2739	0.3045	0.4715
WFG5	0.0886	0.1543	0.2739	0.3047	0.4711
	0.0001	0.0001	0.0002	0.0004	0.0005
	0.0888	0.1546	0.2741	0.3057	0.4725
	0.0884	0.1540	0.2734	0.3042	0.4696

Table D.17: Average, Standard Deviation, Maximum, and Minimum IGD for the NSGA-III (continue)

NSGA-III IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0875	0.1536	0.2732	0.3045	0.4713
	0.0001	0.0001	0.0002	0.0005	0.0006
	0.0877	0.1538	0.2735	0.3058	0.4723
	0.0872	0.1534	0.2728	0.3036	0.4702
WFG7	0.0879	0.1543	0.2739	0.3041	0.4706
	0.0001	0.0002	0.0002	0.0003	0.0005
	0.0882	0.1552	0.2741	0.3051	0.4715
	0.0878	0.1539	0.2736	0.3036	0.4698
WFG8	0.0888	0.1540	0.2736	0.3038	0.4706
	0.0001	0.0001	0.0002	0.0003	0.0005
	0.0890	0.1544	0.2745	0.3046	0.4715
	0.0885	0.1538	0.2733	0.3035	0.4699
WFG9	0.0887	0.1535	0.2731	0.3040	0.4707
	0.0002	0.0002	0.0002	0.0004	0.0005
	0.0893	0.1539	0.2734	0.3049	0.4715
	0.0883	0.1532	0.2727	0.3035	0.4695

Table D.18: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_R$

PMGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0033	0.0076	0.0121	0.0112	0.0121
	0.0002	0.0004	0.0009	0.0008	0.0013
	0.0039	0.0083	0.0136	0.0126	0.0147
	0.0030	0.0068	0.0100	0.0098	0.0094
DTLZ2	0.0047	0.0084	0.0139	0.0145	0.0202
	0.0002	0.0002	0.0003	0.0003	0.0004
	0.0051	0.0087	0.0149	0.0150	0.0210
	0.0045	0.0080	0.0135	0.0140	0.0195
DTLZ3	0.0043	0.0076	0.0136	0.0151	0.0211
	0.0002	0.0002	0.0003	0.0003	0.0003
	0.0048	0.0080	0.0142	0.0158	0.0216
	0.0038	0.0073	0.0129	0.0144	0.0206
DTLZ4	0.0105	0.0235	0.0289	0.0260	0.0279
	0.0006	0.0006	0.0001	0.0001	0.0001
	0.0120	0.0244	0.0292	0.0263	0.0282
	0.0095	0.0219	0.0287	0.0258	0.0277
DTLZ5	0.0048	0.0060	0.0064	0.0070	0.0073
	0.0005	0.0006	0.0005	0.0005	0.0006
	0.0064	0.0070	0.0076	0.0080	0.0089
	0.0040	0.0046	0.0054	0.0062	0.0063
DTLZ6	0.0015	0.0039	0.0052	0.0063	0.0070
	0.0001	0.0005	0.0006	0.0006	0.0007
	0.0019	0.0048	0.0065	0.0079	0.0078
	0.0013	0.0030	0.0043	0.0051	0.0057

Table D.18: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_R$ (continue)

PMGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.0968	0.1306	0.2240	0.0922	0.1374
	0.0004	0.0003	0.0002	0.0001	0.0001
	0.0979	0.1310	0.2244	0.0924	0.1376
	0.0963	0.1300	0.2235	0.0921	0.1372
WFG1	0.0495	0.0612	0.0879	0.0843	0.1336
	0.0002	0.0001	0.0001	0.0001	0.0001
	0.0502	0.0615	0.0882	0.0846	0.1338
	0.0493	0.0609	0.0877	0.0841	0.1334
WFG2	0.0439	0.0574	0.0871	0.0839	0.1334
	0.0004	0.0002	0.0002	0.0002	0.0002
	0.0446	0.0578	0.0874	0.0843	0.1337
	0.0429	0.0568	0.0866	0.0834	0.1331
WFG3	0.0725	0.1396	0.2429	0.3117	0.4844
	0.0003	0.0002	0.0001	0.0001	0.0001
	0.0730	0.1398	0.2431	0.3120	0.4846
	0.0720	0.1392	0.2428	0.3115	0.4843
WFG4	0.0882	0.1558	0.2780	0.3086	0.4763
	0.0002	0.0002	0.0002	0.0003	0.0003
	0.0885	0.1562	0.2787	0.3093	0.4768
	0.0879	0.1553	0.2776	0.3083	0.4758
WFG5	0.0871	0.1537	0.2737	0.3037	0.4697
	0.0002	0.0001	0.0001	0.0002	0.0002
	0.0875	0.1540	0.2740	0.3042	0.4702
	0.0868	0.1534	0.2734	0.3034	0.4693

Table D.18: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_R$ (continue)

PMGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0875	0.1551	0.2756	0.3057	0.4729
	0.0002	0.0001	0.0001	0.0001	0.0002
	0.0879	0.1554	0.2759	0.3059	0.4734
	0.0872	0.1548	0.2754	0.3055	0.4725
WFG7	0.0884	0.1565	0.2774	0.3073	0.4727
	0.0002	0.0002	0.0001	0.0002	0.0003
	0.0886	0.1568	0.2779	0.3077	0.4733
	0.0881	0.1561	0.2772	0.3068	0.4722
WFG8	0.0879	0.1558	0.2765	0.3062	0.4734
	0.0003	0.0003	0.0002	0.0002	0.0002
	0.0885	0.1563	0.2769	0.3065	0.4739
	0.0873	0.1552	0.2761	0.3059	0.4728
WFG9	0.0868	0.1533	0.2729	0.3030	0.4674
	0.0002	0.0001	0.0001	0.0001	0.0002
	0.0871	0.1536	0.2731	0.3033	0.4678
	0.0864	0.1531	0.2727	0.3028	0.4670

Table D.19: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{RI}$

PMGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0033	0.0076	0.0123	0.0113	0.0120
	0.0002	0.0003	0.0008	0.0008	0.0011
	0.0039	0.0083	0.0136	0.0128	0.0147
	0.0029	0.0068	0.0100	0.0093	0.0094
DTLZ2	0.0047	0.0084	0.0139	0.0145	0.0202
	0.0002	0.0002	0.0003	0.0003	0.0004
	0.0051	0.0090	0.0149	0.0151	0.0214
	0.0044	0.0080	0.0133	0.0140	0.0195
DTLZ3	0.0043	0.0077	0.0136	0.0150	0.0211
	0.0002	0.0002	0.0003	0.0003	0.0003
	0.0048	0.0081	0.0144	0.0158	0.0218
	0.0038	0.0073	0.0129	0.0143	0.0202
DTLZ4	0.0106	0.0234	0.0290	0.0260	0.0279
	0.0006	0.0006	0.0002	0.0001	0.0001
	0.0120	0.0245	0.0294	0.0263	0.0283
	0.0095	0.0219	0.0287	0.0258	0.0276
DTLZ5	0.0048	0.0060	0.0063	0.0069	0.0073
	0.0005	0.0006	0.0005	0.0005	0.0007
	0.0064	0.0072	0.0076	0.0080	0.0089
	0.0039	0.0046	0.0052	0.0060	0.0055
DTLZ6	0.0015	0.0038	0.0051	0.0063	0.0069
	0.0001	0.0004	0.0006	0.0006	0.0007
	0.0019	0.0048	0.0065	0.0079	0.0081
	0.0012	0.0030	0.0040	0.0047	0.0052

Table D.19: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{RI}$ (continue)

PMGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.0968	0.1305	0.2240	0.0922	0.1373
	0.0004	0.0003	0.0002	0.0001	0.0001
	0.0986	0.1312	0.2244	0.0924	0.1376
	0.0959	0.1296	0.2233	0.0920	0.1370
WFG1	0.0495	0.0613	0.0879	0.0843	0.1336
	0.0002	0.0002	0.0001	0.0001	0.0001
	0.0502	0.0617	0.0883	0.0846	0.1338
	0.0493	0.0609	0.0876	0.0841	0.1333
WFG2	0.0439	0.0573	0.0871	0.0839	0.1335
	0.0004	0.0002	0.0002	0.0002	0.0002
	0.0448	0.0578	0.0875	0.0843	0.1339
	0.0428	0.0567	0.0866	0.0834	0.1331
WFG3	0.0724	0.1396	0.2429	0.3117	0.4844
	0.0002	0.0002	0.0001	0.0001	0.0001
	0.0730	0.1399	0.2431	0.3120	0.4846
	0.0720	0.1392	0.2428	0.3114	0.4843
WFG4	0.0882	0.1558	0.2780	0.3087	0.4763
	0.0001	0.0002	0.0002	0.0002	0.0003
	0.0885	0.1562	0.2787	0.3093	0.4769
	0.0878	0.1553	0.2776	0.3082	0.4758
WFG5	0.0871	0.1538	0.2737	0.3038	0.4696
	0.0002	0.0001	0.0001	0.0002	0.0002
	0.0875	0.1541	0.2740	0.3042	0.4702
	0.0868	0.1534	0.2734	0.3034	0.4692

Table D.19: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{RI}$ (continue)

PMGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0875	0.1551	0.2756	0.3057	0.4729
	0.0002	0.0001	0.0001	0.0001	0.0002
	0.0880	0.1554	0.2759	0.3060	0.4734
	0.0871	0.1548	0.2754	0.3054	0.4722
WFG7	0.0884	0.1565	0.2775	0.3073	0.4727
	0.0002	0.0002	0.0002	0.0002	0.0002
	0.0888	0.1568	0.2779	0.3077	0.4733
	0.0881	0.1561	0.2772	0.3068	0.4722
WFG8	0.0879	0.1557	0.2765	0.3062	0.4734
	0.0003	0.0003	0.0002	0.0002	0.0002
	0.0885	0.1563	0.2770	0.3065	0.4739
	0.0871	0.1552	0.2760	0.3059	0.4728
WFG9	0.0867	0.1533	0.2729	0.3030	0.4673
	0.0002	0.0001	0.0001	0.0001	0.0002
	0.0871	0.1536	0.2731	0.3033	0.4678
	0.0863	0.1529	0.2726	0.3028	0.4667

Table D.20: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{STD}$

PMGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0033	0.0075	0.0122	0.0111	0.0119
	0.0002	0.0003	0.0010	0.0010	0.0012
	0.0039	0.0083	0.0138	0.0128	0.0147
	0.0028	0.0068	0.0077	0.0063	0.0089
DTLZ2	0.0047	0.0084	0.0140	0.0145	0.0202
	0.0002	0.0002	0.0003	0.0003	0.0004
	0.0052	0.0090	0.0149	0.0153	0.0214
	0.0044	0.0080	0.0133	0.0140	0.0195
DTLZ3	0.0043	0.0077	0.0136	0.0150	0.0211
	0.0002	0.0002	0.0003	0.0003	0.0003
	0.0048	0.0084	0.0144	0.0158	0.0218
	0.0038	0.0073	0.0129	0.0143	0.0202
DTLZ4	0.0107	0.0234	0.0289	0.0260	0.0279
	0.0006	0.0007	0.0002	0.0001	0.0002
	0.0124	0.0245	0.0294	0.0263	0.0283
	0.0095	0.0211	0.0286	0.0257	0.0274
DTLZ5	0.0047	0.0060	0.0064	0.0069	0.0072
	0.0005	0.0006	0.0005	0.0005	0.0006
	0.0064	0.0075	0.0076	0.0080	0.0089
	0.0039	0.0046	0.0049	0.0058	0.0055
DTLZ6	0.0015	0.0038	0.0051	0.0062	0.0069
	0.0002	0.0004	0.0006	0.0006	0.0008
	0.0019	0.0048	0.0065	0.0079	0.0091
	0.0010	0.0030	0.0040	0.0047	0.0051

Table D.20: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{STD}$ (continue)

PMGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.0969	0.1305	0.2240	0.0922	0.1374
	0.0004	0.0003	0.0002	0.0001	0.0001
	0.0986	0.1317	0.2245	0.0925	0.1376
	0.0959	0.1296	0.2233	0.0920	0.1369
WFG1	0.0495	0.0613	0.0879	0.0843	0.1336
	0.0002	0.0002	0.0001	0.0001	0.0001
	0.0503	0.0617	0.0883	0.0848	0.1339
	0.0491	0.0609	0.0876	0.0840	0.1333
WFG2	0.0438	0.0573	0.0871	0.0839	0.1335
	0.0004	0.0002	0.0002	0.0002	0.0002
	0.0448	0.0578	0.0875	0.0843	0.1339
	0.0426	0.0567	0.0866	0.0834	0.1331
WFG3	0.0724	0.1396	0.2430	0.3117	0.4844
	0.0002	0.0001	0.0001	0.0001	0.0001
	0.0730	0.1399	0.2431	0.3120	0.4846
	0.0720	0.1392	0.2427	0.3114	0.4843
WFG4	0.0882	0.1558	0.2780	0.3086	0.4763
	0.0001	0.0002	0.0002	0.0002	0.0003
	0.0885	0.1562	0.2787	0.3093	0.4769
	0.0878	0.1553	0.2776	0.3082	0.4757
WFG5	0.0870	0.1538	0.2737	0.3038	0.4696
	0.0002	0.0001	0.0001	0.0002	0.0002
	0.0875	0.1541	0.2740	0.3042	0.4702
	0.0865	0.1534	0.2733	0.3034	0.4691

Table D.20: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{STD}$ (continue)

PMGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0875	0.1551	0.2756	0.3057	0.4729
	0.0002	0.0001	0.0001	0.0001	0.0002
	0.0880	0.1554	0.2759	0.3060	0.4734
	0.0871	0.1547	0.2753	0.3054	0.4722
WFG7	0.0884	0.1565	0.2775	0.3072	0.4727
	0.0002	0.0002	0.0002	0.0002	0.0002
	0.0889	0.1568	0.2779	0.3077	0.4733
	0.0881	0.1561	0.2771	0.3068	0.4722
WFG8	0.0879	0.1557	0.2765	0.3063	0.4735
	0.0003	0.0003	0.0002	0.0002	0.0002
	0.0889	0.1563	0.2770	0.3067	0.4740
	0.0871	0.1550	0.2760	0.3058	0.4728
WFG9	0.0867	0.1533	0.2729	0.3030	0.4673
	0.0002	0.0001	0.0001	0.0001	0.0002
	0.0871	0.1536	0.2731	0.3033	0.4678
	0.0863	0.1529	0.2726	0.3028	0.4667

Appendix E

Performance Measure Values for Chapter 5

This appendix provides the average, standard deviation, maximum, and minimum HV and IGD performance measure values for each algorithm on each problem instance for Chapter 5. Section E.1 lists the HV performance measure tables; that is, tables E.1 to E.10. Section E.2 lists the IGD performance measure tables; that is, tables E.11 to E.20. Note that the tables are listed alphabetically according to algorithm name. Also, note that some of the algorithms had no valid solutions left over after outlier removal; that is, no solutions in any of the independent samples for that specific problem. In these rare cases no performance measure value could be calculated; indicated with "-".

E.1 Hypervolume Values

The average, standard deviation, maximum, and minimum HV performance measure values for each algorithm on each problem instance are listed in tables E.1 to E.10. Note that these performance measure values are associated with Chapter 5.

Table E.1: Average, Standard Deviation, Maximum, and Minimum HV for the CDAS-SMPSO algorithm

CDAS-SMPSO HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.0941	1.2817	1.4167	1.7194	3.3188
	0.0486	0.0827	0.2052	0.4640	0.4119
	1.1766	1.4209	1.8138	2.4583	3.9321
	0.9934	1.0753	0.9296	0.6545	2.2346
DTLZ2	1.2243	1.4855	1.9212	2.2898	3.0435
	0.0511	0.0440	0.0660	0.0977	0.3960
	1.2845	1.5472	2.0267	2.4558	3.7286
	1.0922	1.3863	1.8188	2.0307	1.6853
DTLZ3	1.2189	1.3641	1.7461	2.2471	3.4356
	0.1106	0.0659	0.0926	0.2288	0.3836
	1.3155	1.6096	1.8919	2.5491	3.7275
	0.8201	1.2178	1.4001	1.6172	2.1305
DTLZ4	1.0878	1.4515	1.7830	1.9503	2.7113
	0.1719	0.1227	0.2550	0.3885	0.5150
	1.2851	1.5678	2.0827	2.5701	3.8472
	0.7033	1.1385	1.0120	1.1566	1.6532
DTLZ5	1.0250	1.2324	1.4294	1.3960	1.8080
	0.0591	0.0829	0.1732	0.1087	0.3269
	1.1664	1.4190	1.7924	1.7150	2.5238
	0.9388	1.1079	1.1511	1.2393	1.1624
DTLZ6	0.9103	1.1034	1.5296	1.9142	3.0994
	0.0895	0.1115	0.1693	0.1512	0.4337
	1.0760	1.3248	1.9013	2.2552	3.9437
	0.7181	0.7987	1.2924	1.6357	2.1727

Table E.1: Average, Standard Deviation, Maximum, and Minimum HV for the CDAS-SMPSO algorithm (continue)

CDAS-SMPSO HV					
Benchmark Function	n_m				
	3	5	8	10	15
DTLZ7	0.0973	0.0363	0.0024	0.0017	0.0010
	0.1143	0.0660	0.0078	0.0071	0.0033
	0.3913	0.2787	0.0422	0.0391	0.0150
	0.0141	0.0004	0.0000	0.0000	0.0000
WFG1	1.0194	1.1905	0.5574	0.2986	0.0166
	0.0526	0.0397	0.1280	0.1523	0.0071
	1.0606	1.2460	0.8675	0.6118	0.0222
	0.8621	1.0688	0.4181	0.0894	0.0062
WFG2	0.9680	1.0257	0.7053	0.4518	1.7194
	0.0310	0.0445	0.1129	0.2085	0.6063
	1.0401	1.0976	0.8923	0.8964	2.6557
	0.8805	0.9262	0.4078	0.0619	0.2073
WFG3	0.7521	0.7549	0.6782	0.7866	1.8335
	0.0213	0.0191	0.0384	0.0528	0.1254
	0.7988	0.7923	0.7645	0.8858	2.0427
	0.7050	0.7064	0.5931	0.6842	1.5362
WFG4	0.7458	0.9165	1.0255	1.0900	1.6461
	0.0153	0.0417	0.0351	0.1035	0.2800
	0.7735	1.0156	1.1050	1.3017	1.9732
	0.7185	0.8302	0.9485	0.8384	0.7266
WFG5	0.7369	0.8234	0.8922	0.7547	0.6917
	0.0202	0.0330	0.0454	0.0765	0.2511
	0.7763	0.8916	0.9949	0.9142	1.1190
	0.7079	0.7678	0.7871	0.5851	0.2417

Table E.1: Average, Standard Deviation, Maximum, and Minimum HV for the CDAS-SMPSO algorithm (continue)

CDAS-SMPSO HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6739	0.7361	0.7455	0.7234	0.9328
	0.0231	0.0292	0.0392	0.0691	0.2262
	0.7007	0.8116	0.8297	0.9039	1.3526
	0.6067	0.6841	0.6663	0.5430	0.2910
WFG7	0.7783	0.8453	0.8506	0.8201	1.3267
	0.0118	0.0309	0.0326	0.0769	0.1734
	0.7992	0.8953	0.9237	0.9620	1.5891
	0.7496	0.7837	0.7883	0.6800	0.9408
WFG8	0.7522	0.7547	0.7900	0.7531	1.2953
	0.0167	0.0378	0.0345	0.0694	0.1447
	0.7837	0.8037	0.8506	0.8715	1.5829
	0.7156	0.6046	0.7190	0.6007	0.9480
WFG9	0.8125	0.8365	0.8896	0.7665	1.2477
	0.0295	0.0306	0.0731	0.2133	0.2562
	0.8782	0.8991	0.9976	1.0648	1.6825
	0.7442	0.7713	0.6241	0.2041	0.2899

Table E.2: Average, Standard Deviation, Maximum, and Minimum HV for the KnEA

KnEA HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3204	-	2.0657	2.4574	3.9287
	0.0013	-	0.0394	0.0781	0.1549
	1.3226	-	2.1084	2.5722	4.1215
	1.3167	-	1.8940	2.2504	3.6251
DTLZ2	1.2694	1.5341	1.5939	2.1676	2.2516
	0.0056	0.0755	0.2512	0.2427	0.6605
	1.2796	1.5992	1.9268	2.5194	3.5368
	1.2573	1.3120	1.0863	1.6222	1.5373
DTLZ3	1.3118	1.4498	1.8570	2.2247	3.7817
	0.0112	0.1473	0.2269	0.2989	0.3399
	1.3190	1.5941	2.0599	2.5266	4.1265
	1.2542	1.0769	1.2762	1.0583	2.8645
DTLZ4	1.2456	1.2123	1.3460	1.6821	2.4787
	0.0578	0.2203	0.0424	0.0430	0.0726
	1.2839	1.5410	1.4051	1.7442	2.5925
	0.9872	0.9473	1.2364	1.5277	2.3423
DTLZ5	1.0110	1.0012	1.0994	1.0426	1.4565
	0.1263	0.0984	0.0697	0.1491	0.3465
	1.1251	1.4249	1.2030	1.6182	2.5582
	0.8004	0.8591	0.9409	0.7472	1.0598
DTLZ6	1.0190	0.6180	0.6955	0.9507	1.5787
	0.0173	0.2074	0.2049	0.1928	0.4509
	1.0429	1.2334	1.1712	1.5353	2.6329
	0.9885	0.4179	0.4935	0.7375	1.0292

Table E.2: Average, Standard Deviation, Maximum, and Minimum HV for the KnEA (continue)

KnEA HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.3713	0.8387	0.9364	0.5275	0.1051
	0.0395	0.0400	0.1736	0.1605	0.0580
	0.4032	0.9303	1.2319	0.9730	0.2435
	0.1714	0.7738	0.4949	0.2984	0.0110
WFG1	0.8071	0.9341	0.6571	0.6027	0.0683
	0.0521	0.1881	0.1728	0.1400	0.0194
	1.0353	1.2399	0.9954	0.8966	0.1046
	0.7283	0.6636	0.4340	0.3476	0.0401
WFG2	0.9421	0.9838	0.6932	0.6601	1.8837
	0.0607	0.0725	0.0909	0.1491	0.5389
	1.0114	1.1255	0.9428	0.9745	2.7972
	0.8456	0.8254	0.5068	0.3481	0.8076
WFG3	0.7028	0.6161	0.4548	0.3814	0.4419
	0.0291	0.0365	0.0529	0.1094	0.4225
	0.7461	0.6849	0.5628	0.6594	1.5560
	0.6398	0.5425	0.3469	0.2150	0.0041
WFG4	0.7835	1.0124	0.7814	0.9472	0.7251
	0.0140	0.0362	0.1239	0.1284	0.2874
	0.8062	1.0794	1.0050	1.1623	1.3178
	0.7328	0.9414	0.4321	0.6454	0.2606
WFG5	0.7990	0.9866	0.5221	0.8136	0.6276
	0.0135	0.0399	0.1937	0.1543	0.2713
	0.8372	1.0499	0.8459	1.0764	1.1634
	0.7799	0.8860	0.2011	0.4489	0.1513

Table E.2: Average, Standard Deviation, Maximum, and Minimum HV for the KnEA (continue)

KnEA HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.7152	0.8661	0.4991	0.7077	0.6349
	0.0172	0.0385	0.1533	0.1367	0.2831
	0.7639	0.9322	0.8058	1.0005	1.3844
	0.6907	0.7562	0.1649	0.4762	0.2280
WFG7	0.7964	0.9727	0.6043	0.6263	0.3819
	0.0149	0.0381	0.0937	0.1406	0.2780
	0.8221	1.0508	0.7905	0.9627	1.3023
	0.7664	0.9009	0.4255	0.3578	0.0692
WFG8	0.7970	0.8532	0.5060	0.5224	0.5932
	0.0155	0.0654	0.1240	0.1733	0.4385
	0.8326	0.9890	0.7001	0.8768	1.3684
	0.7506	0.6932	0.2713	0.2178	0.0615
WFG9	0.7779	0.8666	0.6273	0.6883	0.6861
	0.0258	0.0486	0.1078	0.1619	0.4042
	0.8529	1.0154	0.8998	1.0845	1.5077
	0.7265	0.7824	0.4732	0.3372	0.1022

Table E.3: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_R$

KnMGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.2986	1.4954	1.9681	2.2516	3.7483
	0.0063	0.0163	0.0277	0.0442	0.0609
	1.3144	1.5208	2.0173	2.3508	3.8895
	1.2872	1.4537	1.8934	2.1706	3.6244
DTLZ2	1.3036	1.5672	2.0450	2.3813	3.3631
	0.0012	0.0060	0.0157	0.0382	0.1339
	1.3055	1.5823	2.0719	2.4396	3.5903
	1.3002	1.5540	2.0065	2.3158	3.1155
DTLZ3	1.2895	1.4683	1.8577	2.1216	3.4617
	0.0051	0.0215	0.0388	0.0918	0.1197
	1.2980	1.5175	1.9368	2.2738	3.7277
	1.2786	1.4300	1.7678	1.9358	3.2303
DTLZ4	1.1547	1.5025	1.4499	1.6694	2.5645
	0.1643	0.1253	0.0956	0.0679	0.0450
	1.3119	1.5872	1.5243	1.7482	2.6543
	0.9590	1.1702	1.0169	1.4702	2.4530
DTLZ5	1.1918	1.4489	1.8472	1.9361	2.6921
	0.0013	0.0034	0.0072	0.0168	0.0390
	1.1935	1.4558	1.8605	1.9611	2.7703
	1.1887	1.4426	1.8342	1.9050	2.6177
DTLZ6	0.7637	1.0623	1.3049	1.3415	1.8013
	0.0079	0.0125	0.0284	0.0712	0.1631
	0.7802	1.0826	1.3539	1.5230	2.0673
	0.7527	1.0340	1.2420	1.2106	1.4164

Table E.3: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_R$ (continue)

KnMGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.1215	1.3200	1.2956	1.0280	1.2287
	0.0551	0.0263	0.0665	0.1224	0.2104
	1.3000	1.3917	1.4305	1.2530	1.7945
	0.9748	1.2808	1.1840	0.8371	0.7448
WFG1	0.5108	0.5300	0.6787	0.6570	3.3568
	0.0227	0.0345	0.0682	0.1616	0.6341
	0.5524	0.6045	0.8526	1.0481	3.7126
	0.4624	0.4641	0.5314	0.4841	0.1781
WFG2	0.8410	0.9750	1.2514	1.4601	2.4015
	0.0369	0.0130	0.0204	0.0199	0.0280
	0.8648	1.0031	1.2993	1.4990	2.4613
	0.6545	0.9516	1.2060	1.4145	2.3514
WFG3	0.7827	0.8999	1.1150	1.2656	2.1150
	0.0100	0.0122	0.0159	0.0320	0.0590
	0.8015	0.9272	1.1513	1.3249	2.2094
	0.7616	0.8747	1.0780	1.1935	1.9400
WFG4	0.7703	0.8079	1.0513	1.1847	1.9276
	0.0183	0.0168	0.0210	0.0315	0.0621
	0.8081	0.8360	1.0913	1.2338	2.0496
	0.7290	0.7740	0.9957	1.1088	1.8007
WFG5	0.6871	0.6781	0.7510	0.7528	1.3473
	0.0458	0.0154	0.0265	0.0490	0.0861
	0.7789	0.7126	0.8044	0.8942	1.5216
	0.5870	0.6486	0.7075	0.6543	1.2002

Table E.3: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_R$ (continue)

KnMGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6886	0.7168	0.8210	0.9036	1.4517
	0.0137	0.0117	0.0192	0.0337	0.0417
	0.7077	0.7406	0.8532	0.9666	1.5624
	0.6461	0.6868	0.7468	0.8151	1.3779
WFG7	0.5574	0.6141	0.7329	0.8145	1.2609
	0.0189	0.0169	0.0187	0.0299	0.0472
	0.5904	0.6614	0.7747	0.8825	1.3588
	0.5081	0.5873	0.6937	0.7558	1.1158
WFG8	0.5235	0.5802	0.7004	0.7993	1.2701
	0.0065	0.0133	0.0172	0.0261	0.0443
	0.5356	0.6149	0.7301	0.8591	1.3576
	0.5091	0.5539	0.6611	0.7414	1.1670
WFG9	0.9509	0.9635	1.1211	1.2420	1.9128
	0.0140	0.0228	0.0333	0.0636	0.0883
	0.9756	1.0248	1.2020	1.3951	2.1089
	0.9163	0.9239	1.0439	1.1013	1.7583

Table E.4: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO_{RI}

KnMGPSO_{RI} HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3003	1.4947	1.9631	2.2506	3.7456
	0.0064	0.0138	0.0258	0.0477	0.0593
	1.3202	1.5208	2.0173	2.3598	3.8895
	1.2872	1.4537	1.8901	2.1203	3.6244
DTLZ2	1.3036	1.5668	2.0486	2.3862	3.3825
	0.0012	0.0065	0.0158	0.0398	0.1236
	1.3056	1.5823	2.0792	2.4459	3.6089
	1.3001	1.5505	2.0065	2.2424	3.1155
DTLZ3	1.2905	1.4672	1.8581	2.1242	3.4407
	0.0053	0.0188	0.0423	0.0930	0.1293
	1.3083	1.5175	1.9626	2.3109	3.7277
	1.2786	1.4300	1.7470	1.8980	3.1399
DTLZ4	1.1448	1.5048	1.4600	1.6737	2.5644
	0.1654	0.1092	0.0718	0.0678	0.0430
	1.3119	1.5872	1.5243	1.7547	2.6543
	0.9590	1.1702	1.0169	1.4277	2.4530
DTLZ5	1.1919	1.4485	1.8482	1.9328	2.6913
	0.0011	0.0033	0.0072	0.0174	0.0408
	1.1937	1.4558	1.8619	1.9656	2.7947
	1.1887	1.4418	1.8342	1.9009	2.6177
DTLZ6	0.7631	1.0650	1.3069	1.3406	1.8168
	0.0077	0.0118	0.0290	0.0636	0.1441
	0.7802	1.0864	1.3601	1.5230	2.0673
	0.7489	1.0340	1.2354	1.2106	1.4164

Table E.4: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{RI}$ (continue)

KnMGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.1186	1.3235	1.2890	1.0040	1.2130
	0.0577	0.0309	0.0583	0.1052	0.2001
	1.3000	1.3941	1.4305	1.2530	1.7945
	0.9748	1.2525	1.1679	0.8318	0.7448
WFG1	0.5131	0.5371	0.6731	0.6458	3.2994
	0.0227	0.0386	0.0667	0.1448	0.7268
	0.5714	0.6820	0.8526	1.0481	3.8296
	0.4624	0.4641	0.5314	0.4393	0.1570
WFG2	0.8452	0.9744	1.2549	1.4574	2.4014
	0.0267	0.0129	0.0181	0.0485	0.0311
	0.8648	1.0031	1.2993	1.5176	2.4613
	0.6545	0.9499	1.2060	1.1422	2.3021
WFG3	0.7824	0.8987	1.1133	1.2681	2.1252
	0.0115	0.0109	0.0158	0.0328	0.0673
	0.8074	0.9272	1.1513	1.3249	2.2498
	0.7529	0.8747	1.0780	1.1916	1.9400
WFG4	0.7709	0.8046	1.0475	1.1790	1.9302
	0.0192	0.0191	0.0221	0.0353	0.0578
	0.8142	0.8360	1.0913	1.2523	2.0496
	0.7290	0.7426	0.9930	1.0893	1.8007
WFG5	0.6810	0.6783	0.7447	0.7476	1.3314
	0.0441	0.0219	0.0309	0.0449	0.0976
	0.7789	0.7326	0.8119	0.8942	1.5511
	0.5870	0.6191	0.6847	0.6543	1.1515

Table E.4: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{RI}$ (continue)

KnMGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6897	0.7144	0.8218	0.9066	1.4480
	0.0135	0.0142	0.0185	0.0332	0.0460
	0.7085	0.7595	0.8565	0.9666	1.5624
	0.6461	0.6739	0.7468	0.8151	1.3496
WFG7	0.5582	0.6106	0.7364	0.8124	1.2711
	0.0169	0.0174	0.0216	0.0265	0.0446
	0.5906	0.6614	0.7945	0.8825	1.3646
	0.5081	0.5819	0.6749	0.7558	1.1158
WFG8	0.5230	0.5817	0.7018	0.8080	1.2635
	0.0078	0.0141	0.0177	0.0243	0.0433
	0.5412	0.6149	0.7410	0.8631	1.3576
	0.5086	0.5539	0.6554	0.7414	1.1670
WFG9	0.9514	0.9659	1.1183	1.2433	1.9035
	0.0129	0.0215	0.0318	0.0688	0.0802
	0.9756	1.0316	1.2020	1.3951	2.1224
	0.9163	0.9239	1.0439	1.0981	1.7583

Table E.5: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{STD}$

KnMGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3067	1.4985	1.9725	2.2576	3.7469
	0.0118	0.0150	0.0355	0.0535	0.0701
	1.3309	1.5423	2.1074	2.4510	4.0730
	1.2872	1.4537	1.8901	2.1203	3.5366
DTLZ2	1.3034	1.5668	2.0487	2.3857	3.3683
	0.0017	0.0067	0.0169	0.0432	0.1312
	1.3057	1.5823	2.0792	2.4584	3.6089
	1.2968	1.5505	1.9959	2.2424	2.9870
DTLZ3	1.2949	1.4731	1.8679	2.1323	3.4511
	0.0126	0.0216	0.0444	0.0995	0.1400
	1.3306	1.5496	1.9701	2.3595	3.7548
	1.2348	1.4300	1.7470	1.8980	3.1072
DTLZ4	1.1517	1.4935	1.4852	1.6980	2.5892
	0.1636	0.1104	0.0776	0.0677	0.0514
	1.3119	1.5872	1.8693	1.7951	2.6792
	0.9590	1.1702	1.0169	1.4277	2.4530
DTLZ5	1.1912	1.4485	1.8508	1.9362	2.6923
	0.0017	0.0035	0.0085	0.0185	0.0442
	1.1937	1.4570	1.8713	1.9739	2.8190
	1.1848	1.4361	1.8342	1.9000	2.6168
DTLZ6	0.7601	1.0651	1.3113	1.3511	1.8488
	0.0093	0.0124	0.0289	0.0637	0.1484
	0.7825	1.0925	1.3749	1.5230	2.1125
	0.7355	1.0332	1.2354	1.2106	1.4164

Table E.5: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{STD}$ (continue)

KnMGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.1156	1.3316	1.3224	1.0999	1.4694
	0.0598	0.0382	0.0815	0.1795	0.4657
	1.3000	1.4279	1.5195	1.5625	2.7198
	0.9748	1.2505	1.1679	0.8318	0.7448
WFG1	0.4993	0.5189	0.6347	0.6015	3.0150
	0.0291	0.0466	0.0862	0.1373	1.1086
	0.5714	0.6820	0.8526	1.0481	3.8296
	0.4167	0.3783	0.4698	0.3962	0.0733
WFG2	0.8435	0.9701	1.2543	1.4540	2.4032
	0.0223	0.0156	0.0189	0.0540	0.0366
	0.8648	1.0031	1.2993	1.5208	2.4803
	0.6545	0.9317	1.2060	1.1411	2.3021
WFG3	0.7778	0.8933	1.1195	1.2779	2.1193
	0.0136	0.0136	0.0197	0.0376	0.0675
	0.8074	0.9272	1.1776	1.3600	2.2498
	0.7444	0.8396	1.0780	1.1916	1.9400
WFG4	0.7654	0.8016	1.0443	1.1671	1.9223
	0.0236	0.0193	0.0253	0.0384	0.0621
	0.8142	0.8360	1.1014	1.2523	2.0496
	0.6781	0.7426	0.9765	1.0893	1.7606
WFG5	0.6834	0.6810	0.7582	0.7707	1.3673
	0.0503	0.0248	0.0372	0.0538	0.1118
	0.8876	0.7490	0.8583	0.9056	1.6550
	0.5870	0.6191	0.6847	0.6543	1.1515

Table E.5: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{STD}$ (continue)

KnMGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6794	0.7071	0.8162	0.9015	1.4419
	0.0210	0.0198	0.0221	0.0351	0.0459
	0.7085	0.7595	0.8565	0.9666	1.5624
	0.6130	0.6506	0.7468	0.8029	1.3426
WFG7	0.5597	0.6132	0.7446	0.8228	1.2698
	0.0166	0.0187	0.0261	0.0365	0.0451
	0.5932	0.6614	0.8130	0.9270	1.3646
	0.5081	0.5682	0.6749	0.7558	1.1158
WFG8	0.5221	0.5820	0.7061	0.8120	1.2675
	0.0077	0.0136	0.0176	0.0292	0.0488
	0.5412	0.6149	0.7444	0.8788	1.3887
	0.5072	0.5534	0.6554	0.7217	1.0895
WFG9	0.9473	0.9675	1.1210	1.2344	1.9081
	0.0148	0.0218	0.0335	0.0648	0.0801
	0.9756	1.0316	1.2129	1.3951	2.1224
	0.8952	0.9239	1.0439	1.0981	1.7583

Table E.6: Average, Standard Deviation, Maximum, and Minimum HV for the $MGPSO_R$

$MGPSO_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.2971	1.4956	1.9808	2.2813	3.7539
	0.0043	0.0148	0.0268	0.0409	0.0486
	1.3075	1.5260	2.0172	2.3546	3.8483
	1.2895	1.4509	1.8861	2.1740	3.6676
DTLZ2	1.3040	1.5675	2.0523	2.3965	3.3928
	0.0010	0.0059	0.0135	0.0311	0.1154
	1.3058	1.5763	2.0784	2.4546	3.5980
	1.3015	1.5532	2.0231	2.3131	3.1167
DTLZ3	1.2893	1.4643	1.8550	2.1680	3.5192
	0.0048	0.0128	0.0458	0.0982	0.1587
	1.2978	1.4928	1.9720	2.3192	3.7660
	1.2808	1.4357	1.7737	1.8757	3.2273
DTLZ4	1.1263	1.4886	1.4697	1.6680	2.5076
	0.1609	0.1191	0.1164	0.0549	0.0891
	1.3105	1.5892	1.9672	1.7596	2.6329
	0.9590	1.1702	1.1746	1.4759	2.2881
DTLZ5	1.1926	1.4510	1.8536	1.9391	2.6958
	0.0009	0.0040	0.0050	0.0141	0.0380
	1.1941	1.4577	1.8649	1.9801	2.7976
	1.1907	1.4405	1.8458	1.9168	2.6322
DTLZ6	0.7673	1.0669	1.3260	1.3458	1.9174
	0.0089	0.0121	0.0381	0.0648	0.1358
	0.7862	1.0887	1.3999	1.4486	2.1762
	0.7493	1.0405	1.2560	1.1425	1.5751

Table E.6: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_R$ (continue)

MGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.1234	1.3239	1.2659	1.0182	1.2965
	0.0578	0.0422	0.0560	0.1470	0.2150
	1.2050	1.3926	1.3542	1.3383	1.8046
	0.9951	1.2255	1.1536	0.7735	0.8764
WFG1	0.5093	0.5429	0.7189	0.7044	3.4177
	0.0196	0.0498	0.1001	0.1770	0.2962
	0.5455	0.7106	0.9894	1.2410	3.8951
	0.4658	0.4517	0.5323	0.4480	2.7277
WFG2	0.8387	0.9687	1.2527	1.4589	2.3887
	0.0377	0.0298	0.0208	0.0209	0.0332
	0.8637	1.0080	1.2888	1.5012	2.4383
	0.6489	0.8334	1.2140	1.4175	2.3154
WFG3	0.7750	0.9000	1.1126	1.2500	2.0973
	0.0157	0.0093	0.0227	0.0322	0.0496
	0.8096	0.9181	1.1490	1.3114	2.1734
	0.7386	0.8780	1.0593	1.1924	2.0121
WFG4	0.7513	0.7861	1.0364	1.1798	1.9556
	0.0218	0.0221	0.0179	0.0338	0.0503
	0.7912	0.8319	1.0815	1.2360	2.0697
	0.7049	0.7449	1.0070	1.1060	1.8731
WFG5	0.7060	0.7089	0.7658	0.7690	1.3268
	0.0413	0.0264	0.0236	0.0357	0.1030
	0.7864	0.7566	0.8115	0.8402	1.5099
	0.6375	0.6497	0.7216	0.6973	1.1170

Table E.6: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_R$ (continue)

MGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6914	0.7193	0.8243	0.8953	1.4473
	0.0153	0.0178	0.0253	0.0291	0.0429
	0.7139	0.7551	0.8683	0.9507	1.5194
	0.6469	0.6792	0.7730	0.8327	1.3444
WFG7	0.5683	0.6082	0.7320	0.8131	1.2669
	0.0112	0.0107	0.0227	0.0235	0.0429
	0.5876	0.6372	0.8016	0.8617	1.3812
	0.5446	0.5888	0.6820	0.7622	1.1997
WFG8	0.5256	0.5800	0.6983	0.8012	1.2313
	0.0069	0.0130	0.0223	0.0265	0.0482
	0.5407	0.6063	0.7382	0.8477	1.3258
	0.5131	0.5554	0.6545	0.7439	1.1338
WFG9	0.9482	0.9840	1.1378	1.2383	1.9208
	0.0194	0.0316	0.0382	0.0655	0.1013
	0.9862	1.0902	1.2384	1.4074	2.0635
	0.9022	0.9274	1.0713	1.0978	1.7161

Table E.7: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{RI}$

MGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.2964	1.4959	1.9793	2.2752	3.7526
	0.0043	0.0142	0.0256	0.0405	0.0500
	1.3087	1.5260	2.0188	2.3546	3.8642
	1.2895	1.4509	1.8861	2.1740	3.6178
DTLZ2	1.3041	1.5674	2.0557	2.4090	3.4094
	0.0011	0.0071	0.0139	0.0326	0.1203
	1.3062	1.5812	2.0784	2.4739	3.6238
	1.3011	1.5485	2.0231	2.3131	3.0735
DTLZ3	1.2892	1.4615	1.8650	2.1586	3.5350
	0.0048	0.0165	0.0487	0.0907	0.1378
	1.2978	1.4998	1.9803	2.3195	3.8767
	1.2765	1.4124	1.7487	1.8757	3.2273
DTLZ4	1.1244	1.4917	1.4636	1.6723	2.5188
	0.1577	0.1166	0.1367	0.0500	0.0769
	1.3114	1.5894	1.9672	1.7596	2.6329
	0.9590	1.1702	1.0378	1.4759	2.2881
DTLZ5	1.1927	1.4503	1.8530	1.9403	2.7012
	0.0008	0.0066	0.0065	0.0128	0.0368
	1.1941	1.4577	1.8666	1.9801	2.7976
	1.1907	1.4087	1.8360	1.9168	2.6322
DTLZ6	0.7665	1.0664	1.3216	1.3554	1.8985
	0.0084	0.0116	0.0353	0.0602	0.1311
	0.7864	1.0960	1.3999	1.4771	2.1762
	0.7493	1.0405	1.2336	1.1425	1.5751

Table E.7: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{RI}$ (continue)

MGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.1152	1.3332	1.2683	0.9892	1.2662
	0.0550	0.0393	0.0543	0.1299	0.2230
	1.2050	1.4191	1.3909	1.3383	1.8046
	0.9951	1.2255	1.1536	0.7279	0.7360
WFG1	0.5135	0.5479	0.7384	0.7014	3.3864
	0.0220	0.0529	0.1216	0.2096	0.2758
	0.5564	0.7106	1.2122	1.6224	3.8951
	0.4633	0.4517	0.5323	0.4480	2.7277
WFG2	0.8442	0.9637	1.2551	1.4643	2.3923
	0.0276	0.0477	0.0194	0.0196	0.0339
	0.8662	1.0080	1.2978	1.5012	2.4817
	0.6489	0.7393	1.1996	1.4175	2.3154
WFG3	0.7772	0.8964	1.1137	1.2536	2.1126
	0.0144	0.0112	0.0209	0.0344	0.0544
	0.8129	0.9181	1.1652	1.3387	2.2208
	0.7386	0.8681	1.0593	1.1863	1.9947
WFG4	0.7535	0.7876	1.0385	1.1815	1.9504
	0.0200	0.0211	0.0229	0.0312	0.0511
	0.7912	0.8319	1.0818	1.2541	2.0697
	0.7029	0.7449	0.9884	1.1060	1.8268
WFG5	0.7054	0.7056	0.7699	0.7700	1.3232
	0.0414	0.0269	0.0301	0.0401	0.1064
	0.7864	0.7657	0.8568	0.8786	1.5099
	0.6175	0.6497	0.7216	0.6948	1.0948

Table E.7: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{RI}$ (continue)

MGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6928	0.7209	0.8247	0.9028	1.4432
	0.0132	0.0191	0.0240	0.0310	0.0500
	0.7167	0.7614	0.8683	0.9697	1.5416
	0.6469	0.6642	0.7730	0.8327	1.2947
WFG7	0.5681	0.6109	0.7302	0.8125	1.2622
	0.0135	0.0145	0.0253	0.0246	0.0448
	0.5988	0.6650	0.8114	0.8683	1.3812
	0.5382	0.5888	0.6736	0.7546	1.1509
WFG8	0.5245	0.5806	0.6988	0.7966	1.2358
	0.0076	0.0128	0.0211	0.0330	0.0494
	0.5431	0.6150	0.7382	0.8477	1.3258
	0.5086	0.5554	0.6545	0.6637	1.1336
WFG9	0.9501	0.9788	1.1301	1.2402	1.9087
	0.0166	0.0282	0.0372	0.0650	0.1040
	0.9862	1.0902	1.2384	1.4074	2.0888
	0.9022	0.9092	1.0713	1.0432	1.6335

Table E.8: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{STD}$

MGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3040	1.5012	1.9879	2.2832	3.7610
	0.0124	0.0188	0.0307	0.0505	0.0538
	1.3291	1.5834	2.1295	2.4531	3.8642
	1.2895	1.4509	1.8861	2.1740	3.6178
DTLZ2	1.3035	1.5670	2.0541	2.4072	3.3942
	0.0018	0.0072	0.0153	0.0367	0.1172
	1.3062	1.5833	2.0791	2.4804	3.6238
	1.2957	1.5485	2.0005	2.2867	3.0735
DTLZ3	1.2954	1.4716	1.8779	2.1650	3.5462
	0.0113	0.0269	0.0535	0.0924	0.1421
	1.3264	1.5922	2.1251	2.3460	3.9395
	1.2765	1.4124	1.7487	1.8757	3.2273
DTLZ4	1.1278	1.4668	1.5051	1.6930	2.5567
	0.1569	0.1297	0.1468	0.0542	0.0847
	1.3114	1.5894	1.9672	1.7964	2.6795
	0.9590	1.1700	1.0378	1.4759	2.2881
DTLZ5	1.1920	1.4501	1.8539	1.9420	2.7035
	0.0013	0.0059	0.0068	0.0183	0.0437
	1.1941	1.4577	1.8746	2.0010	2.8301
	1.1876	1.4087	1.8360	1.8986	2.6027
DTLZ6	0.7638	1.0648	1.3239	1.3648	1.9051
	0.0094	0.0116	0.0332	0.0579	0.1325
	0.7864	1.0960	1.3999	1.5035	2.1762
	0.7438	1.0405	1.2336	1.1425	1.5751

Table E.8: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{STD}$ (continue)

MGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.1078	1.3310	1.2968	1.1093	1.4855
	0.0560	0.0379	0.0796	0.2303	0.4298
	1.2050	1.4191	1.5493	1.8103	2.6843
	0.9948	1.2255	1.0987	0.7279	0.7360
WFG1	0.5007	0.5290	0.6817	0.6345	3.2519
	0.0279	0.0552	0.1328	0.2000	0.6149
	0.5564	0.7106	1.2122	1.6224	3.8951
	0.4439	0.4244	0.4511	0.3673	0.2085
WFG2	0.8408	0.9608	1.2542	1.4591	2.3867
	0.0284	0.0468	0.0179	0.0421	0.0356
	0.8662	1.0080	1.2978	1.5177	2.4817
	0.6489	0.7382	1.1996	1.1385	2.3154
WFG3	0.7722	0.8908	1.1170	1.2513	2.1085
	0.0165	0.0145	0.0237	0.0431	0.0590
	0.8129	0.9181	1.1672	1.3508	2.2208
	0.7349	0.8449	1.0513	1.1494	1.9523
WFG4	0.7513	0.7856	1.0358	1.1740	1.9385
	0.0226	0.0222	0.0241	0.0316	0.0557
	0.7923	0.8387	1.0818	1.2541	2.0697
	0.6986	0.7211	0.9823	1.1060	1.7902
WFG5	0.7115	0.7059	0.7803	0.7866	1.3665
	0.0543	0.0293	0.0339	0.0463	0.1253
	0.9094	0.7974	0.9074	0.9030	1.6628
	0.6175	0.6497	0.7216	0.6948	1.0948

Table E.8: Average, Standard Deviation, Maximum, and Minimum HV for the MGPSO$_{STD}$ (continue)

MGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6846	0.7146	0.8189	0.8989	1.4299
	0.0182	0.0214	0.0272	0.0326	0.0541
	0.7167	0.7614	0.8683	0.9960	1.5416
	0.6327	0.6501	0.7465	0.8300	1.2504
WFG7	0.5661	0.6111	0.7336	0.8165	1.2598
	0.0147	0.0164	0.0286	0.0287	0.0472
	0.6011	0.6650	0.8114	0.8977	1.3963
	0.5271	0.5828	0.6725	0.7546	1.1509
WFG8	0.5223	0.5799	0.7015	0.8082	1.2404
	0.0087	0.0129	0.0217	0.0364	0.0514
	0.5431	0.6150	0.7546	0.8815	1.3679
	0.5024	0.5525	0.6545	0.6637	1.1336
WFG9	0.9465	0.9796	1.1269	1.2374	1.9043
	0.0171	0.0294	0.0342	0.0654	0.1003
	0.9862	1.0902	1.2384	1.4074	2.0888
	0.9022	0.9092	1.0568	1.0432	1.6335

Table E.9: Average, Standard Deviation, Maximum, and Minimum HV for the MOEA/DD

MOEA/DD HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3113	1.5804	2.0418	2.2454	3.8671
	0.0026	0.0088	0.0314	0.1112	0.2200
	1.3158	1.6002	2.1044	2.4835	4.0922
	1.3028	1.5609	1.9648	2.0727	2.9342
DTLZ2	1.2400	1.5823	2.0466	1.8116	1.7342
	0.0094	0.0064	0.0232	0.3530	0.7580
	1.2600	1.5927	2.0963	2.4257	3.2490
	1.2168	1.5681	1.9954	1.0625	0.6005
DTLZ3	1.2904	1.5849	2.0916	2.3829	3.9084
	0.0059	0.0077	0.0179	0.1399	0.1592
	1.3000	1.6006	2.1217	2.5458	4.1037
	1.2778	1.5706	2.0553	1.7597	3.5378
DTLZ4	1.2402	1.5958	2.1083	2.1748	2.2679
	0.0151	0.0066	0.0337	0.4036	0.0000
	1.2712	1.6039	2.1415	2.5809	2.2679
	1.2108	1.5750	1.9732	1.2824	2.2679
DTLZ5	1.0677	1.3550	1.4940	1.0242	1.2879
	0.0098	0.0167	0.1770	0.2418	0.2680
	1.0925	1.3819	1.7355	1.6695	2.4431
	1.0444	1.3196	1.2011	0.5881	0.8739
DTLZ6	0.9726	1.3326	1.6600	0.8106	0.6773
	0.0141	0.0138	0.0544	0.3634	0.6836
	0.9963	1.3637	1.7685	1.3343	2.1841
	0.9387	1.3069	1.5328	0.0878	0.1489

Table E.9: Average, Standard Deviation, Maximum, and Minimum HV for the MOEA/DD (continue)

MOEA/DD HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.3815	0.7886	0.2905	0.0254	0.0026
	0.0197	0.0981	0.2867	0.0601	0.0054
	0.4195	0.9595	1.1049	0.2422	0.0242
	0.3454	0.6015	0.0030	0.0001	0.0000
WFG1	1.0276	1.1818	0.5786	0.5626	0.0327
	0.0409	0.0593	0.0941	0.0817	0.0000
	1.0698	1.2550	0.8061	0.7570	0.0327
	0.9277	0.9882	0.4675	0.4576	0.0327
WFG2	0.9080	0.8634	0.4565	0.2755	1.4486
	0.0397	0.0761	0.0813	0.1187	0.5160
	0.9557	0.9840	0.6023	0.7367	2.4893
	0.7775	0.7201	0.2301	0.1241	0.0605
WFG3	0.6375	0.5784	0.4169	0.1417	0.2523
	0.0223	0.0231	0.0487	0.0373	0.2397
	0.6848	0.6332	0.5132	0.2602	0.9288
	0.5811	0.5472	0.3338	0.0854	0.0007
WFG4	0.6741	0.8951	0.9374	0.9987	1.9033
	0.0199	0.0438	0.0318	0.0725	0.1856
	0.7101	0.9733	1.0155	1.1107	2.2662
	0.6363	0.8076	0.8720	0.7562	1.4929
WFG5	0.7108	0.8589	0.8295	0.3661	1.2225
	0.0150	0.0283	0.0552	0.1803	0.5333
	0.7403	0.9077	0.9107	0.8415	2.1637
	0.6774	0.8030	0.6752	0.0999	0.4127

Table E.9: Average, Standard Deviation, Maximum, and Minimum HV for the MOEA/DD (continue)

MOEA/DD HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6346	0.7643	0.7219	0.4335	1.0567
	0.0209	0.0253	0.0372	0.1380	0.4185
	0.6664	0.8327	0.8018	0.6830	1.9528
	0.5857	0.7056	0.6443	0.1529	0.1751
WFG7	0.7000	0.8690	0.7859	0.6611	1.5943
	0.0131	0.0318	0.0436	0.1431	0.4144
	0.7278	0.9359	0.8643	0.9016	2.2333
	0.6686	0.8052	0.6925	0.3632	0.3574
WFG8	0.7180	0.8034	0.7623	0.3964	1.3210
	0.0150	0.0363	0.0493	0.2255	0.4402
	0.7563	0.8723	0.8751	0.8081	2.0524
	0.6931	0.6794	0.6742	0.0771	0.5845
WFG9	0.7166	0.7758	0.7489	0.6768	1.6904
	0.0429	0.0425	0.0604	0.0992	0.3925
	0.8012	0.8853	0.8687	0.8697	2.2785
	0.6541	0.6732	0.6331	0.3905	0.3939

Table E.10: Average, Standard Deviation, Maximum, and Minimum HV for the NSGA-III

NSGA-III HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3117	1.5837	2.0467	2.2924	3.5823
	0.0024	0.0052	0.0231	0.1878	0.2595
	1.3164	1.5966	2.1021	2.4614	3.8891
	1.3071	1.5735	2.0023	1.5699	3.0202
DTLZ2	1.2252	1.5568	1.9916	2.2019	2.6027
	0.0098	0.0142	0.0527	0.2880	0.4380
	1.2413	1.5758	2.0926	2.4618	3.3597
	1.2025	1.5081	1.8707	1.1474	1.5604
DTLZ3	1.2975	1.5932	2.0765	2.4280	3.8088
	0.0040	0.0055	0.0195	0.0893	0.1647
	1.3043	1.6005	2.1108	2.5637	4.0712
	1.2892	1.5792	2.0204	2.2463	3.4639
DTLZ4	1.1474	1.5272	1.9720	2.0879	1.8883
	0.0910	0.0697	0.1384	0.4162	0.1995
	1.2294	1.5864	2.1159	2.5415	2.3132
	0.8550	1.3213	1.5874	0.9714	1.7039
DTLZ5	1.0414	1.2624	1.4586	0.6885	1.0844
	0.0172	0.0736	0.1689	0.1964	0.1955
	1.0689	1.3572	1.6576	1.2784	1.3708
	0.9932	1.0601	1.0025	0.3098	0.6677
DTLZ6	0.9534	1.2815	1.5468	1.4838	1.5716
	0.0192	0.0162	0.0698	0.2768	0.5072
	0.9985	1.3050	1.6816	1.9114	2.3948
	0.9158	1.2419	1.3857	0.5712	0.2084

Table E.10: Average, Standard Deviation, Maximum, and Minimum HV for the NSGA-III (continue)

NSGA-III HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.3201	0.7358	0.8692	0.6842	0.5736
	0.0114	0.0372	0.0897	0.1550	0.1490
	0.3465	0.8257	1.0080	1.0291	0.9209
	0.2929	0.6683	0.6734	0.4115	0.2854
WFG1	0.9867	1.1938	0.6913	0.4976	0.1019
	0.0793	0.0660	0.1756	0.1645	0.1419
	1.0516	1.2561	0.9712	0.7397	0.3556
	0.7331	1.0098	0.3571	0.1481	0.0317
WFG2	0.8674	0.9233	0.6562	0.7056	2.5314
	0.0597	0.0593	0.0538	0.1123	0.3813
	0.9820	1.0317	0.8004	0.9795	3.2615
	0.7975	0.8148	0.5539	0.5297	1.4984
WFG3	0.7233	0.6826	0.5109	0.4787	1.6071
	0.0073	0.0166	0.0400	0.0652	0.2507
	0.7370	0.7159	0.5893	0.6466	2.0853
	0.7085	0.6527	0.4292	0.3922	1.0878
WFG4	0.7211	0.8292	0.9097	0.9637	1.8601
	0.0140	0.0382	0.0292	0.0868	0.1198
	0.7526	0.9324	0.9621	1.1069	2.0818
	0.6941	0.7525	0.8552	0.7647	1.6638
WFG5	0.7274	0.8727	0.8619	0.6595	1.6597
	0.0132	0.0247	0.0378	0.0789	0.1407
	0.7635	0.9239	0.9463	0.8153	1.8621
	0.7013	0.8233	0.8039	0.4917	1.3849

Table E.10: Average, Standard Deviation, Maximum, and Minimum HV for the NSGA-III (continue)

NSGA-III HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.6430	0.7540	0.7434	0.5949	1.6817
	0.0128	0.0267	0.0292	0.0802	0.1521
	0.6712	0.7950	0.8368	0.7588	1.9541
	0.6170	0.6906	0.7000	0.4385	1.4132
WFG7	0.7264	0.8648	0.8374	0.8067	1.7969
	0.0162	0.0491	0.0274	0.0932	0.1550
	0.7575	0.9405	0.8970	1.0035	2.0440
	0.6754	0.7434	0.7823	0.5460	1.3142
WFG8	0.7326	0.7815	0.8241	0.6830	1.6023
	0.0154	0.0252	0.0432	0.0931	0.1129
	0.7585	0.8293	0.9131	0.8680	1.8314
	0.6969	0.7066	0.7437	0.4687	1.4172
WFG9	0.7161	0.7892	0.8566	0.7470	1.9191
	0.0254	0.0481	0.0525	0.0901	0.1436
	0.7989	0.9174	0.9754	0.9103	2.2073
	0.6850	0.6853	0.7393	0.5584	1.6881

E.2 Inverted Generational Distance Values

The average, standard deviation, maximum, and minimum IGD performance measure values for each algorithm on each problem instance are listed in tables E.11 to E.20. Note that these performance measure values are associated with Chapter 5.

Table E.11: Average, Standard Deviation, Maximum, and Minimum IGD for the CDAS-SMPSO algorithm

CDAS-SMPSO IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0069	0.0068	0.0078	0.0078	0.0071
	0.0009	0.0006	0.0012	0.0025	0.0003
	0.0077	0.0076	0.0130	0.0156	0.0079
	0.0045	0.0056	0.0066	0.0059	0.0068
DTLZ2	0.0155	0.0171	0.0203	0.0185	0.0211
	0.0005	0.0006	0.0010	0.0007	0.0009
	0.0165	0.0183	0.0232	0.0200	0.0234
	0.0147	0.0160	0.0191	0.0172	0.0195
DTLZ3	0.0221	0.0256	0.0288	0.0247	0.0265
	0.0020	0.0007	0.0008	0.0012	0.0006
	0.0252	0.0271	0.0292	0.0260	0.0269
	0.0175	0.0231	0.0249	0.0226	0.0248
DTLZ4	0.0187	0.0222	0.0275	0.0254	0.0277
	0.0025	0.0014	0.0013	0.0012	0.0010
	0.0232	0.0255	0.0307	0.0280	0.0301
	0.0129	0.0200	0.0253	0.0225	0.0258

Table E.11: Average, Standard Deviation, Maximum, and Minimum IGD for the CDAS-SMPSO algorithm (continue)

CDAS-SMPSO IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ5	0.0171	0.0185	0.0177	0.0150	0.0144
	0.0003	0.0002	0.0003	0.0003	0.0003
	0.0176	0.0188	0.0181	0.0156	0.0152
	0.0165	0.0181	0.0172	0.0144	0.0137
DTLZ6	0.0230	0.0232	0.0229	0.0238	0.0239
	0.0010	0.0010	0.0015	0.0010	0.0011
	0.0251	0.0254	0.0252	0.0255	0.0262
	0.0215	0.0217	0.0183	0.0222	0.0212
DTLZ7	0.1036	0.1438	0.2399	0.0984	0.1444
	0.0012	0.0018	0.0013	0.0004	0.0005
	0.1061	0.1477	0.2428	0.0993	0.1451
	0.1020	0.1405	0.2363	0.0975	0.1433
WFG1	0.0387	0.0499	0.0819	0.0809	0.1328
	0.0030	0.0019	0.0021	0.0010	0.0004
	0.0462	0.0562	0.0861	0.0824	0.1331
	0.0360	0.0484	0.0786	0.0784	0.1322
WFG2	0.0455	0.0575	0.0925	0.0884	0.1370
	0.0012	0.0010	0.0013	0.0011	0.0007
	0.0482	0.0595	0.0957	0.0915	0.1382
	0.0436	0.0554	0.0906	0.0862	0.1356
WFG3	0.0763	0.1406	0.2428	0.3121	0.4848
	0.0016	0.0007	0.0004	0.0015	0.0010
	0.0798	0.1431	0.2438	0.3168	0.4868
	0.0732	0.1396	0.2419	0.3109	0.4834

Table E.11: Average, Standard Deviation, Maximum, and Minimum IGD for the CDAS-SMPSO algorithm (continue)

CDAS-SMPSO IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG4	0.0877	0.1541	0.2732	0.3042	0.4692
	0.0001	0.0002	0.0002	0.0003	0.0007
	0.0879	0.1545	0.2736	0.3051	0.4706
	0.0875	0.1537	0.2728	0.3036	0.4680
WFG5	0.0882	0.1541	0.2741	0.3047	0.4708
	0.0002	0.0001	0.0002	0.0003	0.0004
	0.0887	0.1543	0.2747	0.3053	0.4717
	0.0879	0.1538	0.2737	0.3041	0.4703
WFG6	0.0877	0.1538	0.2736	0.3045	0.4697
	0.0002	0.0002	0.0003	0.0004	0.0005
	0.0882	0.1541	0.2742	0.3054	0.4714
	0.0871	0.1534	0.2732	0.3039	0.4689
WFG7	0.0879	0.1542	0.2741	0.3039	0.4686
	0.0001	0.0002	0.0002	0.0002	0.0005
	0.0881	0.1546	0.2745	0.3045	0.4698
	0.0878	0.1538	0.2735	0.3035	0.4673
WFG8	0.0881	0.1546	0.2734	0.3037	0.4692
	0.0002	0.0004	0.0001	0.0002	0.0003
	0.0885	0.1563	0.2736	0.3043	0.4700
	0.0878	0.1539	0.2731	0.3034	0.4689
WFG9	0.0894	0.1539	0.2736	0.3052	0.4693
	0.0002	0.0002	0.0006	0.0023	0.0008
	0.0900	0.1543	0.2761	0.3135	0.4720
	0.0888	0.1535	0.2728	0.3034	0.4683

Table E.12: Average, Standard Deviation, Maximum, and Minimum IGD for the KnEA

KnEA IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0015	-	0.0065	0.0058	0.0072
	0.0001	-	0.0006	0.0005	0.0004
	0.0018	-	0.0076	0.0072	0.0082
	0.0013	-	0.0055	0.0048	0.0066
DTLZ2	0.0124	0.0154	0.0269	0.0195	0.0286
	0.0005	0.0023	0.0015	0.0027	0.0017
	0.0137	0.0203	0.0299	0.0263	0.0308
	0.0112	0.0125	0.0236	0.0155	0.0248
DTLZ3	0.0154	0.0213	0.0260	0.0230	0.0254
	0.0006	0.0016	0.0015	0.0017	0.0008
	0.0172	0.0249	0.0293	0.0255	0.0273
	0.0143	0.0187	0.0220	0.0192	0.0231
DTLZ4	0.0124	0.0231	0.0294	0.0263	0.0281
	0.0019	0.0022	0.0001	0.0001	0.0002
	0.0178	0.0254	0.0297	0.0266	0.0284
	0.0104	0.0188	0.0292	0.0261	0.0271
DTLZ5	0.0124	0.0204	0.0200	0.0195	0.0177
	0.0008	0.0017	0.0004	0.0018	0.0030
	0.0140	0.0213	0.0209	0.0208	0.0206
	0.0109	0.0133	0.0187	0.0127	0.0095
DTLZ6	0.0055	0.0224	0.0223	0.0222	0.0216
	0.0004	0.0041	0.0035	0.0029	0.0033
	0.0061	0.0257	0.0257	0.0270	0.0253
	0.0048	0.0128	0.0159	0.0156	0.0153

Table E.12: Average, Standard Deviation, Maximum, and Minimum IGD for the KnEA (continue)

KnEA IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.0992	0.1337	0.2249	0.0933	0.1384
	0.0003	0.0012	0.0013	0.0006	0.0004
	0.0997	0.1358	0.2268	0.0943	0.1395
	0.0986	0.1307	0.2215	0.0920	0.1374
WFG1	0.0471	0.0591	0.0862	0.0832	0.1329
	0.0026	0.0014	0.0022	0.0013	0.0011
	0.0488	0.0611	0.0894	0.0854	0.1347
	0.0358	0.0561	0.0815	0.0805	0.1300
WFG2	0.0445	0.0607	0.0893	0.0852	0.1355
	0.0014	0.0009	0.0010	0.0008	0.0010
	0.0467	0.0627	0.0922	0.0873	0.1381
	0.0405	0.0584	0.0879	0.0839	0.1335
WFG3	0.0781	0.1480	0.2486	0.3178	0.4875
	0.0017	0.0014	0.0024	0.0030	0.0014
	0.0816	0.1510	0.2531	0.3244	0.4905
	0.0752	0.1456	0.2434	0.3125	0.4849
WFG4	0.0878	0.1545	0.2759	0.3049	0.4731
	0.0001	0.0002	0.0015	0.0007	0.0015
	0.0880	0.1551	0.2804	0.3067	0.4770
	0.0877	0.1542	0.2740	0.3038	0.4702
WFG5	0.0886	0.1549	0.2811	0.3070	0.4721
	0.0001	0.0003	0.0034	0.0019	0.0016
	0.0888	0.1562	0.2876	0.3109	0.4769
	0.0882	0.1546	0.2755	0.3041	0.4698

Table E.12: Average, Standard Deviation, Maximum, and Minimum IGD for the KnEA (continue)

KnEA IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0878	0.1546	0.2814	0.3084	0.4736
	0.0001	0.0013	0.0028	0.0015	0.0029
	0.0882	0.1606	0.2871	0.3119	0.4822
	0.0876	0.1539	0.2756	0.3046	0.4701
WFG7	0.0879	0.1545	0.2800	0.3066	0.4703
	0.0001	0.0002	0.0018	0.0021	0.0011
	0.0882	0.1556	0.2839	0.3114	0.4729
	0.0877	0.1542	0.2765	0.3036	0.4682
WFG8	0.0884	0.1549	0.2797	0.3075	0.4712
	0.0001	0.0008	0.0021	0.0024	0.0011
	0.0886	0.1569	0.2849	0.3123	0.4737
	0.0882	0.1542	0.2763	0.3031	0.4694
WFG9	0.0888	0.1542	0.2767	0.3049	0.4696
	0.0003	0.0004	0.0017	0.0013	0.0015
	0.0895	0.1553	0.2818	0.3086	0.4741
	0.0882	0.1536	0.2736	0.3031	0.4675

Table E.13: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_R$

KnMGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0012	0.0068	0.0099	0.0105	0.0113
	0.0001	0.0004	0.0006	0.0010	0.0009
	0.0014	0.0077	0.0110	0.0121	0.0132
	0.0010	0.0060	0.0084	0.0082	0.0092
DTLZ2	0.0157	0.0134	0.0158	0.0152	0.0202
	0.0004	0.0004	0.0004	0.0004	0.0004
	0.0164	0.0142	0.0164	0.0161	0.0212
	0.0150	0.0123	0.0146	0.0146	0.0196
DTLZ3	0.0096	0.0086	0.0141	0.0152	0.0211
	0.0005	0.0004	0.0004	0.0002	0.0006
	0.0107	0.0095	0.0149	0.0158	0.0224
	0.0085	0.0079	0.0135	0.0148	0.0202
DTLZ4	0.0205	0.0227	0.0291	0.0263	0.0279
	0.0024	0.0011	0.0004	0.0002	0.0001
	0.0232	0.0256	0.0308	0.0268	0.0282
	0.0178	0.0215	0.0288	0.0261	0.0277
DTLZ5	0.0173	0.0146	0.0126	0.0092	0.0082
	0.0004	0.0005	0.0006	0.0007	0.0005
	0.0178	0.0154	0.0135	0.0104	0.0093
	0.0165	0.0136	0.0114	0.0080	0.0074
DTLZ6	0.0017	0.0040	0.0052	0.0064	0.0078
	0.0002	0.0006	0.0007	0.0006	0.0010
	0.0022	0.0053	0.0066	0.0076	0.0095
	0.0012	0.0031	0.0041	0.0053	0.0061

Table E.13: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_R$ (continue)

KnMGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1173	0.1373	0.2261	0.0926	0.1378
	0.0011	0.0015	0.0005	0.0002	0.0002
	0.1193	0.1407	0.2272	0.0930	0.1385
	0.1133	0.1330	0.2248	0.0923	0.1373
WFG1	0.0499	0.0615	0.0882	0.0845	0.1337
	0.0002	0.0003	0.0002	0.0003	0.0003
	0.0507	0.0623	0.0889	0.0852	0.1347
	0.0496	0.0612	0.0879	0.0841	0.1335
WFG2	0.0490	0.0608	0.0899	0.0857	0.1356
	0.0009	0.0004	0.0002	0.0002	0.0003
	0.0518	0.0620	0.0906	0.0863	0.1360
	0.0474	0.0604	0.0895	0.0853	0.1351
WFG3	0.0746	0.1409	0.2434	0.3120	0.4842
	0.0001	0.0001	0.0000	0.0001	0.0001
	0.0747	0.1410	0.2435	0.3121	0.4844
	0.0745	0.1407	0.2433	0.3118	0.4841
WFG4	0.0886	0.1559	0.2779	0.3086	0.4771
	0.0001	0.0001	0.0002	0.0003	0.0003
	0.0887	0.1561	0.2783	0.3092	0.4776
	0.0885	0.1557	0.2773	0.3079	0.4765
WFG5	0.0878	0.1538	0.2737	0.3040	0.4701
	0.0004	0.0001	0.0001	0.0001	0.0003
	0.0886	0.1541	0.2740	0.3043	0.4709
	0.0869	0.1535	0.2735	0.3037	0.4697

Table E.13: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_R$ (continue)

KnMGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0905	0.1564	0.2759	0.3058	0.4730
	0.0001	0.0002	0.0002	0.0002	0.0003
	0.0908	0.1567	0.2761	0.3062	0.4734
	0.0902	0.1561	0.2756	0.3055	0.4722
WFG7	0.0895	0.1568	0.2777	0.3074	0.4728
	0.0001	0.0001	0.0002	0.0002	0.0003
	0.0898	0.1571	0.2780	0.3078	0.4736
	0.0892	0.1566	0.2770	0.3069	0.4723
WFG8	0.0898	0.1570	0.2769	0.3064	0.4736
	0.0001	0.0002	0.0002	0.0002	0.0003
	0.0901	0.1574	0.2774	0.3070	0.4741
	0.0897	0.1567	0.2766	0.3059	0.4731
WFG9	0.0905	0.1546	0.2733	0.3033	0.4678
	0.0002	0.0002	0.0001	0.0002	0.0003
	0.0906	0.1551	0.2737	0.3037	0.4684
	0.0898	0.1542	0.2731	0.3030	0.4671

Table E.14: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{RI}$

KnMGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0011	0.0067	0.0098	0.0105	0.0113
	0.0001	0.0004	0.0006	0.0010	0.0007
	0.0016	0.0077	0.0110	0.0123	0.0132
	0.0010	0.0059	0.0084	0.0079	0.0092
DTLZ2	0.0159	0.0134	0.0159	0.0151	0.0202
	0.0005	0.0004	0.0004	0.0003	0.0004
	0.0167	0.0142	0.0166	0.0161	0.0212
	0.0150	0.0123	0.0146	0.0146	0.0194
DTLZ3	0.0096	0.0086	0.0142	0.0153	0.0211
	0.0006	0.0003	0.0004	0.0003	0.0005
	0.0109	0.0095	0.0150	0.0162	0.0226
	0.0085	0.0079	0.0131	0.0146	0.0201
DTLZ4	0.0206	0.0227	0.0291	0.0263	0.0279
	0.0024	0.0010	0.0003	0.0002	0.0001
	0.0232	0.0256	0.0308	0.0270	0.0282
	0.0172	0.0215	0.0288	0.0260	0.0274
DTLZ5	0.0173	0.0146	0.0126	0.0092	0.0082
	0.0004	0.0006	0.0006	0.0006	0.0005
	0.0178	0.0159	0.0138	0.0104	0.0094
	0.0165	0.0130	0.0114	0.0080	0.0072
DTLZ6	0.0017	0.0039	0.0052	0.0064	0.0075
	0.0002	0.0005	0.0006	0.0007	0.0009
	0.0022	0.0053	0.0066	0.0080	0.0095
	0.0012	0.0031	0.0039	0.0051	0.0056

Table E.14: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{RI}$ (continue)

KnMGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1170	0.1374	0.2260	0.0926	0.1378
	0.0015	0.0017	0.0005	0.0002	0.0002
	0.1197	0.1420	0.2272	0.0932	0.1385
	0.1120	0.1330	0.2248	0.0922	0.1373
WFG1	0.0499	0.0615	0.0882	0.0845	0.1337
	0.0003	0.0002	0.0003	0.0003	0.0003
	0.0508	0.0623	0.0891	0.0852	0.1351
	0.0496	0.0611	0.0879	0.0841	0.1335
WFG2	0.0489	0.0608	0.0898	0.0857	0.1356
	0.0007	0.0004	0.0002	0.0003	0.0002
	0.0518	0.0620	0.0906	0.0872	0.1360
	0.0474	0.0604	0.0893	0.0853	0.1351
WFG3	0.0746	0.1409	0.2434	0.3119	0.4842
	0.0001	0.0001	0.0000	0.0001	0.0001
	0.0748	0.1410	0.2435	0.3121	0.4845
	0.0744	0.1407	0.2433	0.3117	0.4841
WFG4	0.0886	0.1559	0.2779	0.3086	0.4771
	0.0001	0.0001	0.0002	0.0003	0.0003
	0.0887	0.1562	0.2784	0.3093	0.4777
	0.0885	0.1557	0.2773	0.3079	0.4765
WFG5	0.0878	0.1538	0.2737	0.3040	0.4701
	0.0004	0.0002	0.0001	0.0002	0.0002
	0.0886	0.1543	0.2741	0.3043	0.4709
	0.0869	0.1534	0.2735	0.3036	0.4696

Table E.14: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{RI}$ (continue)

KnMGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0905	0.1564	0.2759	0.3058	0.4730
	0.0001	0.0002	0.0002	0.0002	0.0002
	0.0908	0.1568	0.2762	0.3062	0.4734
	0.0902	0.1558	0.2756	0.3055	0.4722
WFG7	0.0895	0.1569	0.2777	0.3074	0.4728
	0.0001	0.0001	0.0002	0.0002	0.0003
	0.0898	0.1571	0.2780	0.3078	0.4736
	0.0892	0.1566	0.2770	0.3068	0.4722
WFG8	0.0898	0.1570	0.2769	0.3064	0.4736
	0.0001	0.0002	0.0002	0.0002	0.0003
	0.0901	0.1574	0.2774	0.3070	0.4743
	0.0896	0.1566	0.2764	0.3059	0.4731
WFG9	0.0905	0.1546	0.2733	0.3034	0.4678
	0.0002	0.0002	0.0001	0.0002	0.0002
	0.0907	0.1552	0.2737	0.3037	0.4684
	0.0898	0.1542	0.2730	0.3030	0.4671

Table E.15: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{STD}$

KnMGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0019	0.0066	0.0097	0.0104	0.0112
	0.0016	0.0004	0.0007	0.0010	0.0009
	0.0086	0.0077	0.0110	0.0123	0.0132
	0.0010	0.0054	0.0070	0.0063	0.0069
DTLZ2	0.0159	0.0133	0.0158	0.0151	0.0202
	0.0005	0.0004	0.0004	0.0003	0.0004
	0.0168	0.0142	0.0166	0.0161	0.0215
	0.0148	0.0123	0.0146	0.0144	0.0194
DTLZ3	0.0109	0.0087	0.0142	0.0153	0.0211
	0.0027	0.0004	0.0005	0.0003	0.0006
	0.0224	0.0098	0.0157	0.0163	0.0226
	0.0085	0.0079	0.0131	0.0146	0.0199
DTLZ4	0.0206	0.0228	0.0290	0.0262	0.0278
	0.0024	0.0010	0.0004	0.0002	0.0001
	0.0232	0.0256	0.0308	0.0270	0.0282
	0.0172	0.0215	0.0268	0.0260	0.0273
DTLZ5	0.0172	0.0146	0.0127	0.0092	0.0082
	0.0005	0.0006	0.0006	0.0005	0.0006
	0.0178	0.0159	0.0142	0.0104	0.0098
	0.0159	0.0130	0.0114	0.0078	0.0070
DTLZ6	0.0016	0.0039	0.0052	0.0064	0.0075
	0.0002	0.0005	0.0006	0.0007	0.0010
	0.0022	0.0053	0.0066	0.0080	0.0095
	0.0011	0.0031	0.0039	0.0051	0.0056

Table E.15: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{STD}$ (continue)

KnMGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1165	0.1385	0.2263	0.0927	0.1380
	0.0016	0.0025	0.0008	0.0003	0.0004
	0.1197	0.1467	0.2283	0.0937	0.1398
	0.1119	0.1330	0.2248	0.0922	0.1373
WFG1	0.0497	0.0615	0.0883	0.0846	0.1338
	0.0003	0.0003	0.0004	0.0003	0.0004
	0.0508	0.0625	0.0898	0.0859	0.1351
	0.0492	0.0611	0.0876	0.0841	0.1335
WFG2	0.0489	0.0608	0.0898	0.0857	0.1356
	0.0006	0.0004	0.0002	0.0003	0.0002
	0.0518	0.0620	0.0906	0.0872	0.1361
	0.0474	0.0603	0.0893	0.0852	0.1350
WFG3	0.0746	0.1409	0.2434	0.3119	0.4842
	0.0001	0.0001	0.0000	0.0001	0.0001
	0.0748	0.1410	0.2435	0.3122	0.4845
	0.0743	0.1407	0.2433	0.3117	0.4840
WFG4	0.0886	0.1559	0.2779	0.3087	0.4770
	0.0001	0.0001	0.0002	0.0003	0.0003
	0.0887	0.1563	0.2784	0.3093	0.4777
	0.0884	0.1557	0.2773	0.3079	0.4762
WFG5	0.0877	0.1537	0.2737	0.3040	0.4701
	0.0005	0.0002	0.0001	0.0001	0.0003
	0.0892	0.1543	0.2741	0.3043	0.4709
	0.0866	0.1533	0.2735	0.3036	0.4696

Table E.15: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{STD}$ (continue)

KnMGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0904	0.1563	0.2759	0.3058	0.4730
	0.0002	0.0002	0.0002	0.0002	0.0002
	0.0908	0.1568	0.2762	0.3062	0.4735
	0.0899	0.1558	0.2756	0.3055	0.4722
WFG7	0.0895	0.1569	0.2776	0.3074	0.4728
	0.0001	0.0001	0.0002	0.0002	0.0003
	0.0898	0.1572	0.2780	0.3078	0.4740
	0.0892	0.1564	0.2770	0.3067	0.4722
WFG8	0.0898	0.1570	0.2769	0.3064	0.4736
	0.0001	0.0002	0.0002	0.0002	0.0003
	0.0901	0.1575	0.2777	0.3070	0.4743
	0.0896	0.1566	0.2764	0.3059	0.4730
WFG9	0.0904	0.1546	0.2733	0.3033	0.4678
	0.0002	0.0002	0.0001	0.0002	0.0002
	0.0907	0.1552	0.2737	0.3037	0.4684
	0.0898	0.1542	0.2729	0.3030	0.4671

Table E.16: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_R$

MGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0011	0.0067	0.0097	0.0104	0.0109
	0.0001	0.0003	0.0006	0.0010	0.0007
	0.0014	0.0075	0.0107	0.0125	0.0121
	0.0010	0.0060	0.0086	0.0082	0.0095
DTLZ2	0.0159	0.0136	0.0162	0.0153	0.0206
	0.0004	0.0004	0.0005	0.0005	0.0006
	0.0165	0.0145	0.0172	0.0163	0.0220
	0.0150	0.0125	0.0153	0.0145	0.0197
DTLZ3	0.0095	0.0087	0.0145	0.0155	0.0221
	0.0007	0.0003	0.0006	0.0005	0.0008
	0.0119	0.0093	0.0155	0.0165	0.0233
	0.0088	0.0081	0.0135	0.0145	0.0206
DTLZ4	0.0211	0.0231	0.0289	0.0263	0.0280
	0.0022	0.0011	0.0005	0.0002	0.0002
	0.0232	0.0256	0.0300	0.0268	0.0285
	0.0176	0.0217	0.0269	0.0260	0.0277
DTLZ5	0.0174	0.0149	0.0129	0.0094	0.0084
	0.0003	0.0006	0.0005	0.0005	0.0006
	0.0179	0.0157	0.0136	0.0102	0.0095
	0.0166	0.0134	0.0116	0.0085	0.0074
DTLZ6	0.0016	0.0039	0.0053	0.0065	0.0072
	0.0002	0.0005	0.0005	0.0007	0.0009
	0.0021	0.0050	0.0062	0.0082	0.0097
	0.0014	0.0029	0.0046	0.0052	0.0056

Table E.16: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_R$ (continue)

MGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1173	0.1408	0.2255	0.0925	0.1377
	0.0012	0.0037	0.0005	0.0002	0.0002
	0.1192	0.1470	0.2264	0.0929	0.1383
	0.1149	0.1350	0.2247	0.0922	0.1373
WFG1	0.0503	0.0618	0.0884	0.0846	0.1337
	0.0004	0.0005	0.0004	0.0003	0.0002
	0.0514	0.0642	0.0902	0.0858	0.1342
	0.0497	0.0613	0.0881	0.0842	0.1335
WFG2	0.0492	0.0611	0.0902	0.0859	0.1357
	0.0006	0.0005	0.0003	0.0003	0.0002
	0.0508	0.0622	0.0912	0.0864	0.1361
	0.0482	0.0604	0.0898	0.0854	0.1355
WFG3	0.0746	0.1409	0.2435	0.3120	0.4842
	0.0001	0.0001	0.0001	0.0001	0.0001
	0.0748	0.1410	0.2436	0.3121	0.4843
	0.0745	0.1408	0.2433	0.3118	0.4841
WFG4	0.0886	0.1561	0.2783	0.3089	0.4772
	0.0000	0.0001	0.0002	0.0003	0.0003
	0.0887	0.1563	0.2787	0.3093	0.4779
	0.0886	0.1559	0.2779	0.3084	0.4765
WFG5	0.0880	0.1540	0.2739	0.3040	0.4701
	0.0004	0.0002	0.0002	0.0001	0.0003
	0.0887	0.1544	0.2742	0.3043	0.4707
	0.0872	0.1537	0.2736	0.3037	0.4695

Table E.16: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_R$ (continue)

MGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0907	0.1566	0.2760	0.3059	0.4730
	0.0002	0.0002	0.0002	0.0002	0.0002
	0.0909	0.1569	0.2764	0.3063	0.4734
	0.0902	0.1563	0.2756	0.3055	0.4726
WFG7	0.0897	0.1569	0.2777	0.3074	0.4729
	0.0001	0.0001	0.0002	0.0002	0.0003
	0.0898	0.1572	0.2783	0.3077	0.4735
	0.0895	0.1568	0.2773	0.3069	0.4724
WFG8	0.0899	0.1572	0.2771	0.3066	0.4737
	0.0001	0.0002	0.0002	0.0002	0.0003
	0.0902	0.1576	0.2774	0.3070	0.4745
	0.0897	0.1568	0.2765	0.3063	0.4731
WFG9	0.0904	0.1549	0.2734	0.3033	0.4676
	0.0002	0.0003	0.0001	0.0002	0.0002
	0.0908	0.1555	0.2737	0.3037	0.4681
	0.0898	0.1544	0.2731	0.3030	0.4672

Table E.17: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{RI}$

MGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0012	0.0066	0.0097	0.0106	0.0111
	0.0001	0.0004	0.0005	0.0008	0.0008
	0.0015	0.0075	0.0107	0.0125	0.0124
	0.0010	0.0060	0.0086	0.0082	0.0095
DTLZ2	0.0160	0.0137	0.0163	0.0154	0.0206
	0.0004	0.0005	0.0005	0.0004	0.0006
	0.0166	0.0149	0.0172	0.0163	0.0220
	0.0150	0.0125	0.0153	0.0145	0.0197
DTLZ3	0.0095	0.0086	0.0145	0.0156	0.0221
	0.0007	0.0003	0.0005	0.0005	0.0007
	0.0119	0.0094	0.0159	0.0170	0.0233
	0.0081	0.0080	0.0135	0.0145	0.0206
DTLZ4	0.0212	0.0231	0.0290	0.0263	0.0280
	0.0021	0.0011	0.0006	0.0001	0.0002
	0.0232	0.0256	0.0307	0.0268	0.0285
	0.0176	0.0217	0.0265	0.0260	0.0277
DTLZ5	0.0174	0.0149	0.0128	0.0093	0.0083
	0.0004	0.0006	0.0005	0.0006	0.0006
	0.0179	0.0160	0.0137	0.0104	0.0095
	0.0162	0.0134	0.0116	0.0081	0.0071
DTLZ6	0.0016	0.0038	0.0053	0.0064	0.0071
	0.0002	0.0005	0.0005	0.0007	0.0008
	0.0022	0.0050	0.0064	0.0083	0.0097
	0.0013	0.0027	0.0043	0.0048	0.0055

Table E.17: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{RI}$ (continue)

MGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1171	0.1413	0.2255	0.0925	0.1378
	0.0013	0.0037	0.0004	0.0002	0.0003
	0.1192	0.1470	0.2264	0.0930	0.1384
	0.1141	0.1350	0.2247	0.0922	0.1370
WFG1	0.0502	0.0619	0.0885	0.0846	0.1337
	0.0004	0.0008	0.0004	0.0003	0.0001
	0.0514	0.0648	0.0905	0.0860	0.1342
	0.0497	0.0613	0.0880	0.0841	0.1335
WFG2	0.0493	0.0612	0.0901	0.0858	0.1357
	0.0006	0.0005	0.0003	0.0002	0.0002
	0.0508	0.0626	0.0912	0.0864	0.1362
	0.0482	0.0604	0.0896	0.0854	0.1354
WFG3	0.0747	0.1409	0.2435	0.3120	0.4842
	0.0001	0.0001	0.0001	0.0001	0.0001
	0.0749	0.1411	0.2436	0.3121	0.4844
	0.0745	0.1408	0.2433	0.3118	0.4841
WFG4	0.0886	0.1561	0.2782	0.3088	0.4771
	0.0000	0.0001	0.0002	0.0002	0.0003
	0.0887	0.1565	0.2787	0.3093	0.4779
	0.0885	0.1559	0.2779	0.3084	0.4765
WFG5	0.0880	0.1540	0.2738	0.3040	0.4700
	0.0004	0.0002	0.0002	0.0002	0.0003
	0.0887	0.1544	0.2743	0.3044	0.4707
	0.0872	0.1535	0.2736	0.3036	0.4695

Table E.17: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{RI}$ (continue)

MGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0907	0.1566	0.2760	0.3059	0.4730
	0.0001	0.0002	0.0002	0.0002	0.0002
	0.0909	0.1569	0.2764	0.3063	0.4735
	0.0902	0.1561	0.2756	0.3055	0.4726
WFG7	0.0897	0.1570	0.2777	0.3074	0.4729
	0.0001	0.0001	0.0002	0.0002	0.0003
	0.0899	0.1572	0.2783	0.3079	0.4735
	0.0894	0.1567	0.2773	0.3069	0.4722
WFG8	0.0899	0.1571	0.2771	0.3066	0.4737
	0.0001	0.0002	0.0002	0.0002	0.0003
	0.0902	0.1576	0.2776	0.3070	0.4745
	0.0896	0.1566	0.2765	0.3063	0.4731
WFG9	0.0905	0.1548	0.2733	0.3033	0.4676
	0.0002	0.0003	0.0001	0.0001	0.0002
	0.0908	0.1555	0.2738	0.3037	0.4681
	0.0898	0.1544	0.2731	0.3030	0.4671

Table E.18: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{STD}$

MGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0017	0.0065	0.0097	0.0104	0.0110
	0.0012	0.0004	0.0005	0.0010	0.0008
	0.0069	0.0075	0.0108	0.0125	0.0124
	0.0010	0.0052	0.0077	0.0060	0.0092
DTLZ2	0.0158	0.0136	0.0162	0.0154	0.0206
	0.0005	0.0005	0.0005	0.0004	0.0005
	0.0166	0.0149	0.0175	0.0163	0.0220
	0.0143	0.0124	0.0153	0.0144	0.0197
DTLZ3	0.0107	0.0088	0.0147	0.0156	0.0220
	0.0023	0.0005	0.0014	0.0006	0.0007
	0.0197	0.0124	0.0269	0.0173	0.0235
	0.0081	0.0080	0.0135	0.0145	0.0206
DTLZ4	0.0211	0.0233	0.0288	0.0262	0.0279
	0.0021	0.0011	0.0007	0.0001	0.0002
	0.0232	0.0256	0.0307	0.0268	0.0285
	0.0176	0.0216	0.0265	0.0259	0.0276
DTLZ5	0.0173	0.0148	0.0128	0.0093	0.0084
	0.0005	0.0006	0.0005	0.0006	0.0006
	0.0179	0.0160	0.0138	0.0104	0.0096
	0.0155	0.0134	0.0116	0.0079	0.0071
DTLZ6	0.0016	0.0039	0.0052	0.0064	0.0071
	0.0002	0.0006	0.0005	0.0007	0.0009
	0.0022	0.0057	0.0064	0.0083	0.0097
	0.0012	0.0027	0.0039	0.0048	0.0051

Table E.18: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{STD}$ (continue)

MGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1165	0.1417	0.2258	0.0926	0.1380
	0.0015	0.0035	0.0008	0.0003	0.0005
	0.1192	0.1470	0.2284	0.0934	0.1401
	0.1133	0.1333	0.2247	0.0922	0.1370
WFG1	0.0500	0.0618	0.0885	0.0846	0.1337
	0.0005	0.0007	0.0004	0.0003	0.0003
	0.0514	0.0648	0.0905	0.0860	0.1354
	0.0493	0.0611	0.0880	0.0841	0.1335
WFG2	0.0492	0.0611	0.0901	0.0858	0.1357
	0.0006	0.0005	0.0003	0.0003	0.0002
	0.0508	0.0626	0.0912	0.0872	0.1362
	0.0482	0.0604	0.0896	0.0854	0.1350
WFG3	0.0746	0.1409	0.2435	0.3120	0.4842
	0.0001	0.0001	0.0001	0.0001	0.0001
	0.0749	0.1411	0.2436	0.3122	0.4844
	0.0744	0.1407	0.2432	0.3118	0.4840
WFG4	0.0886	0.1561	0.2782	0.3088	0.4771
	0.0000	0.0001	0.0002	0.0002	0.0003
	0.0888	0.1565	0.2788	0.3093	0.4779
	0.0885	0.1558	0.2778	0.3083	0.4763
WFG5	0.0879	0.1540	0.2738	0.3040	0.4700
	0.0005	0.0002	0.0002	0.0002	0.0003
	0.0895	0.1544	0.2743	0.3044	0.4707
	0.0870	0.1535	0.2736	0.3036	0.4695

Table E.18: Average, Standard Deviation, Maximum, and Minimum IGD for the MGPSO$_{STD}$ (continue)

MGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0906	0.1565	0.2760	0.3059	0.4731
	0.0002	0.0002	0.0002	0.0002	0.0002
	0.0909	0.1569	0.2764	0.3063	0.4735
	0.0899	0.1561	0.2756	0.3055	0.4726
WFG7	0.0896	0.1570	0.2777	0.3074	0.4729
	0.0001	0.0001	0.0002	0.0002	0.0003
	0.0899	0.1572	0.2783	0.3079	0.4737
	0.0892	0.1566	0.2772	0.3069	0.4722
WFG8	0.0899	0.1571	0.2771	0.3066	0.4737
	0.0001	0.0002	0.0002	0.0002	0.0003
	0.0902	0.1576	0.2776	0.3071	0.4746
	0.0896	0.1566	0.2765	0.3063	0.4731
WFG9	0.0904	0.1548	0.2733	0.3033	0.4676
	0.0002	0.0003	0.0001	0.0001	0.0002
	0.0908	0.1557	0.2738	0.3037	0.4681
	0.0897	0.1543	0.2731	0.3030	0.4671

Table E.19: Average, Standard Deviation, Maximum, and Minimum IGD for the MOEA/DD

MOEA/DD IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0009	0.0026	0.0054	0.0054	0.0068
	0.0000	0.0001	0.0003	0.0005	0.0008
	0.0010	0.0030	0.0065	0.0067	0.0103
	0.0008	0.0023	0.0047	0.0047	0.0059
DTLZ2	0.0077	0.0107	0.0165	0.0202	0.0263
	0.0006	0.0005	0.0008	0.0022	0.0022
	0.0090	0.0118	0.0182	0.0254	0.0321
	0.0067	0.0098	0.0147	0.0160	0.0234
DTLZ3	0.0105	0.0129	0.0180	0.0199	0.0231
	0.0004	0.0005	0.0007	0.0010	0.0009
	0.0112	0.0144	0.0195	0.0216	0.0256
	0.0096	0.0121	0.0164	0.0180	0.0209
DTLZ4	0.0071	0.0126	0.0195	0.0224	0.0286
	0.0011	0.0019	0.0018	0.0016	0.0000
	0.0095	0.0164	0.0229	0.0257	0.0286
	0.0055	0.0103	0.0168	0.0193	0.0286
DTLZ5	0.0069	0.0110	0.0112	0.0119	0.0121
	0.0008	0.0006	0.0006	0.0029	0.0009
	0.0080	0.0121	0.0126	0.0191	0.0140
	0.0047	0.0099	0.0101	0.0090	0.0106
DTLZ6	0.0036	0.0068	0.0088	0.0190	0.0274
	0.0003	0.0005	0.0007	0.0052	0.0056
	0.0043	0.0079	0.0101	0.0315	0.0333
	0.0030	0.0057	0.0071	0.0128	0.0153

Table E.19: Average, Standard Deviation, Maximum, and Minimum IGD for the MOEA/DD (continue)

MOEA/DD IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.0997	0.1398	0.2353	0.0980	0.1435
	0.0004	0.0004	0.0008	0.0003	0.0002
	0.1004	0.1407	0.2371	0.0985	0.1442
	0.0991	0.1388	0.2341	0.0972	0.1430
WFG1	0.0389	0.0538	0.0884	0.0845	0.1344
	0.0033	0.0029	0.0006	0.0002	0.0000
	0.0451	0.0577	0.0894	0.0849	0.1344
	0.0348	0.0483	0.0871	0.0839	0.1344
WFG2	0.0437	0.0616	0.0907	0.0883	0.1351
	0.0009	0.0012	0.0021	0.0016	0.0011
	0.0452	0.0648	0.0960	0.0906	0.1372
	0.0410	0.0602	0.0878	0.0850	0.1327
WFG3	0.0768	0.1463	0.2467	0.3186	0.4892
	0.0011	0.0013	0.0028	0.0014	0.0028
	0.0784	0.1486	0.2498	0.3224	0.4989
	0.0739	0.1438	0.2425	0.3130	0.4841
WFG4	0.0880	0.1543	0.2744	0.3051	0.4739
	0.0005	0.0002	0.0004	0.0006	0.0015
	0.0890	0.1551	0.2750	0.3064	0.4782
	0.0873	0.1540	0.2736	0.3042	0.4715
WFG5	0.0880	0.1554	0.2744	0.3103	0.4736
	0.0001	0.0008	0.0005	0.0037	0.0017
	0.0883	0.1572	0.2761	0.3198	0.4771
	0.0878	0.1544	0.2738	0.3041	0.4707

Table E.19: Average, Standard Deviation, Maximum, and Minimum IGD for the MOEA/DD (continue)

MOEA/DD IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0872	0.1543	0.2737	0.3080	0.4733
	0.0002	0.0006	0.0004	0.0029	0.0020
	0.0876	0.1556	0.2751	0.3156	0.4787
	0.0869	0.1536	0.2732	0.3044	0.4703
WFG7	0.0874	0.1550	0.2739	0.3052	0.4714
	0.0002	0.0008	0.0002	0.0012	0.0013
	0.0883	0.1564	0.2743	0.3094	0.4760
	0.0871	0.1540	0.2733	0.3035	0.4698
WFG8	0.0880	0.1542	0.2735	0.3062	0.4720
	0.0002	0.0002	0.0003	0.0028	0.0016
	0.0884	0.1549	0.2742	0.3123	0.4761
	0.0876	0.1538	0.2731	0.3031	0.4695
WFG9	0.0875	0.1535	0.2731	0.3039	0.4707
	0.0004	0.0002	0.0002	0.0007	0.0009
	0.0882	0.1540	0.2736	0.3066	0.4729
	0.0868	0.1531	0.2727	0.3030	0.4679

Table E.20: Average, Standard Deviation, Maximum, and Minimum IGD for the NSGA-III

NSGA-III IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0009	0.0037	0.0075	0.0072	0.0092
	0.0001	0.0003	0.0006	0.0010	0.0009
	0.0010	0.0043	0.0090	0.0106	0.0114
	0.0008	0.0032	0.0066	0.0059	0.0074
DTLZ2	0.0092	0.0090	0.0159	0.0172	0.0224
	0.0005	0.0004	0.0023	0.0014	0.0012
	0.0100	0.0098	0.0222	0.0205	0.0251
	0.0080	0.0084	0.0134	0.0147	0.0203
DTLZ3	0.0128	0.0117	0.0171	0.0174	0.0201
	0.0004	0.0005	0.0018	0.0011	0.0007
	0.0135	0.0126	0.0201	0.0196	0.0222
	0.0122	0.0105	0.0140	0.0154	0.0188
DTLZ4	0.0110	0.0135	0.0207	0.0230	0.0297
	0.0037	0.0038	0.0028	0.0022	0.0009
	0.0166	0.0217	0.0265	0.0271	0.0310
	0.0075	0.0089	0.0156	0.0193	0.0282
DTLZ5	0.0090	0.0102	0.0094	0.0132	0.0107
	0.0006	0.0010	0.0012	0.0035	0.0032
	0.0101	0.0129	0.0121	0.0233	0.0230
	0.0076	0.0084	0.0074	0.0082	0.0071
DTLZ6	0.0039	0.0073	0.0099	0.0152	0.0176
	0.0004	0.0006	0.0019	0.0044	0.0050
	0.0050	0.0081	0.0145	0.0231	0.0284
	0.0033	0.0059	0.0070	0.0080	0.0085

Table E.20: Average, Standard Deviation, Maximum, and Minimum IGD for the NSGA-III (continue)

NSGA-III IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.0976	0.1317	0.2221	0.0930	0.1385
	0.0004	0.0010	0.0008	0.0003	0.0004
	0.0983	0.1337	0.2233	0.0937	0.1394
	0.0966	0.1300	0.2204	0.0925	0.1379
WFG1	0.0393	0.0514	0.0836	0.0815	0.1332
	0.0035	0.0026	0.0028	0.0020	0.0014
	0.0474	0.0563	0.0880	0.0834	0.1348
	0.0350	0.0476	0.0758	0.0765	0.1312
WFG2	0.0433	0.0586	0.0876	0.0843	0.1332
	0.0010	0.0013	0.0010	0.0009	0.0006
	0.0454	0.0603	0.0891	0.0860	0.1342
	0.0414	0.0551	0.0852	0.0823	0.1317
WFG3	0.0733	0.1400	0.2451	0.3169	0.4880
	0.0007	0.0004	0.0013	0.0015	0.0019
	0.0749	0.1409	0.2484	0.3205	0.4909
	0.0722	0.1392	0.2430	0.3130	0.4851
WFG4	0.0874	0.1542	0.2743	0.3049	0.4725
	0.0001	0.0002	0.0002	0.0002	0.0005
	0.0876	0.1550	0.2748	0.3054	0.4737
	0.0873	0.1539	0.2739	0.3045	0.4715
WFG5	0.0880	0.1542	0.2739	0.3047	0.4711
	0.0001	0.0001	0.0002	0.0004	0.0005
	0.0882	0.1545	0.2742	0.3057	0.4725
	0.0878	0.1539	0.2734	0.3041	0.4696

Table E.20: Average, Standard Deviation, Maximum, and Minimum IGD for the NSGA-III (continue)

NSGA-III IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0872	0.1536	0.2732	0.3045	0.4712
	0.0001	0.0001	0.0002	0.0005	0.0006
	0.0874	0.1538	0.2735	0.3058	0.4722
	0.0870	0.1534	0.2728	0.3036	0.4702
WFG7	0.0874	0.1540	0.2737	0.3041	0.4706
	0.0001	0.0002	0.0002	0.0003	0.0005
	0.0877	0.1549	0.2740	0.3051	0.4715
	0.0872	0.1537	0.2734	0.3036	0.4698
WFG8	0.0880	0.1539	0.2735	0.3038	0.4705
	0.0001	0.0001	0.0002	0.0003	0.0005
	0.0882	0.1542	0.2745	0.3046	0.4715
	0.0877	0.1537	0.2732	0.3035	0.4699
WFG9	0.0881	0.1532	0.2730	0.3039	0.4707
	0.0002	0.0002	0.0002	0.0004	0.0005
	0.0888	0.1536	0.2733	0.3049	0.4715
	0.0876	0.1528	0.2726	0.3034	0.4695

Appendix F

Performance Measure Values for Chapter 6

This appendix provides the average, standard deviation, maximum, and minimum HV and IGD performance measure values for each algorithm on each problem instance for Chapter 6. Section F.1 lists the HV performance measure tables; that is, tables F.1 to F.6. Section F.2 lists the IGD performance measure tables; that is, tables F.7 to F.12. Note that the tables are listed alphabetically according to algorithm name.

F.1 Hypervolume Values

The average, standard deviation, maximum, and minimum HV performance measure values for each algorithm on each problem instance are listed in tables F.1 to F.6. Note that these performance measure values are associated with Chapter 6.

Table F.1: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_R$

KnMGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3128	1.5339	1.9658	2.3460	3.8769
	0.0035	0.0120	0.0292	0.0492	0.0458
	1.3215	1.5535	2.0187	2.4340	3.9722
	1.3064	1.5064	1.8908	2.2035	3.7871
DTLZ2	1.3213	1.5884	2.0553	2.3936	3.6928
	0.0005	0.0033	0.0163	0.0419	0.0984
	1.3219	1.5964	2.0816	2.4585	3.8502
	1.3196	1.5813	2.0078	2.3206	3.4986
DTLZ3	1.3043	1.5300	1.9684	2.2668	3.7586
	0.0033	0.0123	0.0284	0.0795	0.0850
	1.3097	1.5575	2.0227	2.3631	3.9242
	1.2972	1.5077	1.8841	2.0902	3.5033
DTLZ4	1.2132	1.4366	1.6303	2.4376	3.9240
	0.1120	0.0591	0.0900	0.0843	0.0993
	1.3193	1.4960	1.7003	2.5355	4.1222
	1.0797	1.2898	1.2226	2.1902	3.6777
DTLZ5	1.2707	1.4918	1.8887	2.2175	3.3282
	0.0006	0.0025	0.0063	0.0105	0.0245
	1.2715	1.4972	1.8993	2.2338	3.3810
	1.2693	1.4867	1.8771	2.1991	3.2796
DTLZ6	0.7795	1.0696	1.2673	1.2702	1.7473
	0.0077	0.0124	0.0286	0.0758	0.1712
	0.7956	1.0891	1.3162	1.4557	2.0098
	0.7687	1.0409	1.2066	1.1426	1.3451

Table F.1: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_R$ (continue)

KnMGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.1238	1.3195	1.3259	1.2377	1.4467
	0.0545	0.0264	0.0851	0.1564	0.2521
	1.3003	1.3913	1.5021	1.5225	2.1225
	0.9787	1.2801	1.1701	0.9940	0.8603
WFG1	0.7997	0.5584	0.5760	0.3148	3.3295
	0.0891	0.1372	0.3509	0.5219	0.6368
	0.9636	0.9260	1.8411	2.3630	3.6117
	0.6296	0.2943	0.1390	0.0173	0.0003
WFG2	1.1746	1.3603	1.7957	2.1131	3.5793
	0.0487	0.0329	0.0307	0.0348	0.0396
	1.2153	1.4362	1.8446	2.1860	3.6359
	0.9590	1.2877	1.7112	2.0361	3.5052
WFG3	1.0489	1.1342	1.3046	1.3936	2.4540
	0.0167	0.0230	0.0351	0.0655	0.1001
	1.0814	1.1830	1.3941	1.5059	2.6431
	1.0116	1.0816	1.2187	1.2844	2.2563
WFG4	0.6977	0.6417	0.6227	0.6261	1.4612
	0.0122	0.0145	0.0244	0.0418	0.0705
	0.7239	0.6715	0.6776	0.7036	1.5703
	0.6715	0.6175	0.5733	0.5374	1.3125
WFG5	0.7571	0.7929	0.7633	0.7320	1.0833
	0.0455	0.0177	0.0309	0.0619	0.0910
	0.8485	0.8334	0.8275	0.9118	1.2645
	0.6573	0.7602	0.7093	0.6091	0.9390

Table F.1: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_R$ (continue)

KnMGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.9104	0.8793	0.9216	0.9654	1.5325
	0.0197	0.0184	0.0308	0.0622	0.0818
	0.9399	0.9150	0.9727	1.0947	1.7286
	0.8499	0.8335	0.8123	0.8330	1.3883
WFG7	0.7426	0.7675	0.8693	0.9211	1.5080
	0.0251	0.0219	0.0228	0.0509	0.0711
	0.7838	0.8243	0.9195	1.0449	1.6677
	0.6806	0.7359	0.8129	0.8279	1.2810
WFG8	0.9058	0.8896	0.9940	1.0573	1.6805
	0.0122	0.0274	0.0368	0.0524	0.1047
	0.9272	0.9638	1.0547	1.1505	1.8872
	0.8794	0.8376	0.9192	0.9376	1.4178
WFG9	0.9709	0.9842	1.1148	1.1988	1.8305
	0.0137	0.0230	0.0354	0.0682	0.0902
	0.9953	1.0453	1.2000	1.3606	2.0322
	0.9370	0.9465	1.0291	1.0410	1.6534

Table F.2: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{RI}$

KnMGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3137	1.5335	1.9603	2.3437	3.8745
	0.0036	0.0107	0.0276	0.0514	0.0458
	1.3249	1.5535	2.0187	2.4340	3.9722
	1.3064	1.5064	1.8752	2.1738	3.7633
DTLZ2	1.3213	1.5882	2.0585	2.3995	3.7038
	0.0005	0.0035	0.0156	0.0444	0.1042
	1.3220	1.5964	2.0844	2.4650	3.8596
	1.3196	1.5791	2.0078	2.2240	3.3158
DTLZ3	1.3049	1.5292	1.9668	2.2720	3.7437
	0.0034	0.0110	0.0336	0.0798	0.0905
	1.3164	1.5575	2.0324	2.4075	3.9242
	1.2972	1.5077	1.8402	2.0172	3.4654
DTLZ4	1.2062	1.4374	1.6397	2.4429	3.9237
	0.1126	0.0513	0.0675	0.0842	0.0949
	1.3193	1.4960	1.7003	2.5435	4.1222
	1.0797	1.2898	1.2226	2.1374	3.6777
DTLZ5	1.2708	1.4916	1.8895	2.2155	3.3279
	0.0005	0.0024	0.0062	0.0111	0.0257
	1.2715	1.4972	1.9012	2.2374	3.3916
	1.2693	1.4867	1.8771	2.1960	3.2796
DTLZ6	0.7789	1.0724	1.2687	1.2681	1.7622
	0.0075	0.0117	0.0291	0.0683	0.1527
	0.7956	1.0939	1.3175	1.4557	2.0125
	0.7651	1.0409	1.1997	1.1426	1.3451

Table F.2: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{RI}$ (continue)

KnMGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.1209	1.3230	1.3172	1.2065	1.4279
	0.0570	0.0310	0.0755	0.1349	0.2402
	1.3003	1.3942	1.5021	1.5225	2.1225
	0.9787	1.2515	1.1555	0.9897	0.8603
WFG1	0.8158	0.5932	0.5609	0.2569	3.2661
	0.0967	0.1754	0.3413	0.4112	0.7641
	1.0669	1.4261	1.8411	2.3630	3.7226
	0.6296	0.2943	0.1390	0.0173	0.0001
WFG2	1.1804	1.3602	1.8009	2.1120	3.5812
	0.0365	0.0318	0.0281	0.0663	0.0464
	1.2196	1.4362	1.8692	2.2024	3.6710
	0.9590	1.2877	1.7112	1.7313	3.4376
WFG3	1.0484	1.1317	1.3026	1.3996	2.4687
	0.0205	0.0220	0.0330	0.0664	0.1084
	1.0867	1.1830	1.3941	1.5185	2.6625
	1.0004	1.0816	1.2108	1.2581	2.2289
WFG4	0.6981	0.6396	0.6197	0.6255	1.4658
	0.0128	0.0154	0.0281	0.0405	0.0676
	0.7270	0.6772	0.6800	0.7115	1.5882
	0.6715	0.6062	0.5618	0.5374	1.3125
WFG5	0.7511	0.7931	0.7555	0.7253	1.0685
	0.0439	0.0255	0.0373	0.0565	0.0972
	0.8485	0.8564	0.8336	0.9118	1.2697
	0.6573	0.7239	0.6902	0.6091	0.9063

Table F.2: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{RI}$ (continue)

KnMGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.9117	0.8759	0.9238	0.9711	1.5248
	0.0197	0.0220	0.0300	0.0590	0.0828
	0.9399	0.9456	0.9869	1.0947	1.7286
	0.8499	0.8138	0.8123	0.8314	1.3595
WFG7	0.7432	0.7636	0.8719	0.9178	1.5227
	0.0225	0.0208	0.0315	0.0459	0.0756
	0.7875	0.8243	0.9247	1.0449	1.6941
	0.6806	0.7294	0.7568	0.8093	1.2810
WFG8	0.9044	0.8930	0.9987	1.0695	1.6658
	0.0153	0.0287	0.0396	0.0488	0.1000
	0.9430	0.9638	1.0892	1.1911	1.8872
	0.8767	0.8375	0.9067	0.9376	1.4178
WFG9	0.9714	0.9867	1.1120	1.1995	1.8204
	0.0126	0.0218	0.0336	0.0736	0.0835
	0.9953	1.0540	1.2000	1.3606	2.0524
	0.9370	0.9465	1.0291	1.0410	1.6534

Table F.3: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{STD}$

KnMGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.3173	1.5367	1.9695	2.3483	3.8743
	0.0066	0.0113	0.0377	0.0539	0.0562
	1.3309	1.5671	2.1102	2.5057	4.1078
	1.3064	1.5064	1.8752	2.1738	3.6786
DTLZ2	1.3211	1.5883	2.0583	2.4003	3.6931
	0.0007	0.0035	0.0172	0.0475	0.1116
	1.3220	1.5964	2.0844	2.4809	3.8596
	1.3183	1.5791	1.9947	2.2240	3.2707
DTLZ3	1.3074	1.5328	1.9740	2.2767	3.7513
	0.0097	0.0125	0.0345	0.0843	0.0938
	1.3307	1.5764	2.0397	2.4439	3.9432
	1.2444	1.5077	1.8402	2.0172	3.4654
DTLZ4	1.2111	1.4304	1.6621	2.4731	3.9785
	0.1114	0.0515	0.0673	0.0841	0.1135
	1.3193	1.4960	1.9084	2.5937	4.1772
	1.0797	1.2898	1.2226	2.1374	3.6777
DTLZ5	1.2705	1.4916	1.8918	2.2188	3.3291
	0.0007	0.0025	0.0074	0.0129	0.0280
	1.2715	1.4978	1.9097	2.2462	3.4111
	1.2676	1.4827	1.8771	2.1960	3.2796
DTLZ6	0.7759	1.0726	1.2734	1.2801	1.7972
	0.0091	0.0122	0.0297	0.0686	0.1569
	0.7978	1.0997	1.3434	1.4557	2.0659
	0.7521	1.0409	1.1997	1.1426	1.3451

Table F.3: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{STD}$ (continue)

KnMGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	1.1179	1.3311	1.3567	1.3279	1.7321
	0.0592	0.0383	0.1032	0.2293	0.5564
	1.3003	1.4275	1.6064	1.9255	3.2312
	0.9787	1.2497	1.1555	0.9897	0.8603
WFG1	0.6895	0.4829	0.4211	0.1813	2.9611
	0.2014	0.2173	0.3464	0.3520	1.1705
	1.0669	1.4261	1.8411	2.3630	3.7226
	0.2481	0.0999	0.0436	0.0046	0.0000
WFG2	1.1744	1.3469	1.7948	2.1006	3.5821
	0.0326	0.0416	0.0321	0.0742	0.0553
	1.2196	1.4362	1.8692	2.2024	3.6973
	0.9590	1.2107	1.7105	1.7262	3.4302
WFG3	1.0395	1.1180	1.3124	1.4250	2.4630
	0.0242	0.0301	0.0388	0.0766	0.1067
	1.0867	1.1830	1.4129	1.5851	2.6625
	0.9810	1.0137	1.2108	1.2581	2.2289
WFG4	0.6948	0.6401	0.6258	0.6265	1.4557
	0.0165	0.0156	0.0314	0.0397	0.0727
	0.7270	0.6772	0.7024	0.7115	1.5882
	0.6287	0.5998	0.5618	0.5374	1.2534
WFG5	0.7541	0.7961	0.7735	0.7569	1.1076
	0.0501	0.0288	0.0459	0.0707	0.1113
	0.9587	0.8768	0.8885	0.9422	1.3793
	0.6573	0.7239	0.6902	0.6091	0.9063

Table F.3: Average, Standard Deviation, Maximum, and Minimum HV for the KnMGPSO$_{STD}$ (continue)

KnMGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.8961	0.8651	0.9187	0.9661	1.5172
	0.0312	0.0296	0.0333	0.0605	0.0820
	0.9399	0.9456	0.9869	1.0947	1.7286
	0.7986	0.7823	0.8123	0.8023	1.3533
WFG7	0.7441	0.7652	0.8772	0.9316	1.5213
	0.0223	0.0219	0.0332	0.0546	0.0785
	0.7903	0.8243	0.9357	1.0747	1.6941
	0.6806	0.7131	0.7568	0.8093	1.2810
WFG8	0.9026	0.8934	1.0085	1.0775	1.6739
	0.0150	0.0277	0.0390	0.0566	0.1086
	0.9430	0.9638	1.0995	1.2150	1.8872
	0.8713	0.8370	0.9067	0.9256	1.2692
WFG9	0.9674	0.9881	1.1154	1.1901	1.8251
	0.0144	0.0221	0.0354	0.0700	0.0838
	0.9953	1.0540	1.2095	1.3606	2.0524
	0.9170	0.9465	1.0291	1.0410	1.6534

Table F.4: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_R$

PMGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.2315	1.4676	1.8499	2.2262	3.5740
	0.0052	0.0127	0.0296	0.0417	0.0844
	1.2402	1.4944	1.9467	2.3213	3.7919
	1.2208	1.4333	1.7910	2.1590	3.3776
DTLZ2	1.2529	1.4911	1.8927	2.1854	3.3265
	0.0093	0.0221	0.0367	0.0737	0.1363
	1.2648	1.5384	1.9464	2.3038	3.5527
	1.2178	1.4584	1.8091	2.0230	3.0707
DTLZ3	1.1372	1.3785	1.7709	2.0091	3.1574
	0.0191	0.0292	0.0430	0.0920	0.2251
	1.1809	1.4438	1.8672	2.2177	3.6398
	1.1053	1.3356	1.7111	1.7895	2.6980
DTLZ4	1.2960	1.2243	1.5929	2.4023	3.8437
	0.0051	0.0274	0.0429	0.0666	0.1314
	1.3062	1.2629	1.6731	2.5216	4.0731
	1.2822	1.1606	1.5103	2.2664	3.5575
DTLZ5	1.2034	1.3571	1.6725	1.9228	2.8920
	0.0101	0.0263	0.0334	0.0476	0.1010
	1.2204	1.3974	1.7542	2.0211	3.0902
	1.1675	1.2678	1.5924	1.8140	2.6462
DTLZ6	0.7533	1.0586	1.2885	1.3341	1.8842
	0.0096	0.0149	0.0306	0.0556	0.1045
	0.7702	1.0900	1.3378	1.4394	2.0820
	0.7240	1.0285	1.2083	1.1921	1.6136

Table F.4: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_R$ (continue)

PMGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.9864	1.1765	1.2531	1.2870	1.4743
	0.0709	0.0722	0.1570	0.1979	0.2852
	1.1210	1.2765	1.4870	1.5813	2.0590
	0.8395	1.0143	0.8595	0.8782	0.7661
WFG1	0.2956	0.1309	0.0273	0.0041	0.0000
	0.0177	0.0135	0.0036	0.0009	0.0000
	0.3251	0.1598	0.0337	0.0061	0.0000
	0.2596	0.1074	0.0204	0.0028	0.0000
WFG2	1.0281	1.1028	1.3512	1.5618	3.0107
	0.0160	0.0316	0.0523	0.0962	0.1720
	1.0570	1.1729	1.4637	1.7336	3.3104
	0.9982	1.0414	1.2426	1.3330	2.7376
WFG3	0.8910	0.9365	1.0898	1.1766	2.2056
	0.0179	0.0300	0.0576	0.0857	0.1299
	0.9247	0.9871	1.2011	1.3403	2.4326
	0.8522	0.8743	0.9449	0.9827	1.9744
WFG4	0.6574	0.6059	0.6086	0.6250	1.3826
	0.0153	0.0156	0.0223	0.0342	0.0623
	0.6866	0.6376	0.6475	0.6835	1.4960
	0.6250	0.5818	0.5738	0.5506	1.2710
WFG5	0.6651	0.7960	0.7904	0.7530	1.1467
	0.0338	0.0310	0.0352	0.0414	0.0611
	0.7403	0.8461	0.8988	0.8132	1.3198
	0.5780	0.7197	0.7320	0.6413	1.0362

Table F.4: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_R$ (continue)

PMGPSO$_R$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.7545	0.7768	0.8635	0.9223	1.4286
	0.0247	0.0313	0.0412	0.0481	0.0980
	0.7950	0.8326	0.9303	1.0084	1.5943
	0.7091	0.7162	0.7690	0.8318	1.1628
WFG7	0.6867	0.7357	0.8475	0.8836	1.4723
	0.0199	0.0189	0.0317	0.0507	0.0841
	0.7459	0.7806	0.8976	0.9728	1.6911
	0.6571	0.7115	0.7570	0.7823	1.2506
WFG8	0.8162	0.8235	0.9468	0.9827	1.5479
	0.0150	0.0348	0.0322	0.0521	0.0848
	0.8385	0.8882	1.0302	1.0729	1.6868
	0.7740	0.7512	0.8851	0.8737	1.3447
WFG9	0.8625	0.8901	1.0576	1.1336	1.8196
	0.0230	0.0217	0.0265	0.0722	0.0893
	0.9118	0.9227	1.1130	1.2594	1.9832
	0.8194	0.8430	1.0028	0.9760	1.6531

Table F.5: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{RI}$

PMGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.2313	1.4687	1.8484	2.2255	3.5653
	0.0059	0.0122	0.0298	0.0398	0.0824
	1.2449	1.4949	1.9467	2.3213	3.7919
	1.2176	1.4333	1.7904	2.1476	3.3471
DTLZ2	1.2525	1.4920	1.8938	2.1889	3.3386
	0.0109	0.0199	0.0387	0.0664	0.1384
	1.2715	1.5411	1.9783	2.3041	3.6129
	1.2178	1.4584	1.8091	2.0230	3.0707
DTLZ3	1.1376	1.3810	1.7725	2.0271	3.1979
	0.0179	0.0294	0.0515	0.0923	0.2039
	1.1809	1.4573	1.9270	2.2667	3.6683
	1.1010	1.3266	1.6780	1.7895	2.6980
DTLZ4	1.2970	1.2277	1.5773	2.4009	3.8325
	0.0046	0.0247	0.0528	0.0605	0.1246
	1.3062	1.2840	1.6731	2.5216	4.0731
	1.2822	1.1606	1.4160	2.2664	3.5558
DTLZ5	1.2024	1.3611	1.6728	1.9284	2.8986
	0.0095	0.0226	0.0359	0.0568	0.0930
	1.2204	1.4065	1.7542	2.0383	3.0902
	1.1675	1.2678	1.5924	1.7919	2.6462
DTLZ6	0.7544	1.0595	1.2897	1.3203	1.8764
	0.0085	0.0133	0.0303	0.0543	0.1042
	0.7702	1.0900	1.3568	1.4394	2.0820
	0.7240	1.0285	1.2083	1.1916	1.6136

Table F.5: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{RI}$ (continue)

PMGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.9878	1.1825	1.2472	1.2411	1.4261
	0.0733	0.0839	0.1575	0.2068	0.2749
	1.1784	1.3232	1.5690	1.5813	2.0590
	0.8395	0.9121	0.8358	0.7338	0.7661
WFG1	0.2967	0.1307	0.0266	0.0042	0.0000
	0.0185	0.0122	0.0038	0.0009	0.0000
	0.3340	0.1598	0.0360	0.0061	0.0000
	0.2565	0.1074	0.0185	0.0028	0.0000
WFG2	1.0270	1.1054	1.3505	1.5651	3.0038
	0.0167	0.0335	0.0554	0.1023	0.1706
	1.0647	1.1925	1.4637	1.7717	3.3403
	0.9944	1.0414	1.2426	1.2986	2.6030
WFG3	0.8910	0.9385	1.1002	1.1824	2.1871
	0.0178	0.0275	0.0522	0.0775	0.1462
	0.9247	0.9874	1.2011	1.3403	2.4326
	0.8522	0.8743	0.9449	0.9827	1.8330
WFG4	0.6556	0.6086	0.6067	0.6144	1.3824
	0.0161	0.0179	0.0238	0.0346	0.0611
	0.6900	0.6501	0.6626	0.7023	1.4960
	0.6130	0.5664	0.5516	0.5506	1.2625
WFG5	0.6612	0.7992	0.7902	0.7632	1.1413
	0.0315	0.0304	0.0356	0.0435	0.0729
	0.7403	0.8642	0.8988	0.8655	1.3198
	0.5780	0.7197	0.7282	0.6413	0.8868

Table F.5: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{RI}$ (continue)

PMGPSO$_{RI}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.7542	0.7791	0.8690	0.9216	1.4429
	0.0278	0.0272	0.0403	0.0477	0.0948
	0.8093	0.8326	0.9419	1.0084	1.6454
	0.7046	0.7036	0.7690	0.8182	1.1628
WFG7	0.6854	0.7353	0.8499	0.8851	1.4708
	0.0187	0.0204	0.0327	0.0518	0.0785
	0.7459	0.7806	0.9142	0.9768	1.6911
	0.6517	0.6925	0.7570	0.7785	1.2506
WFG8	0.8173	0.8175	0.9434	0.9856	1.5625
	0.0175	0.0324	0.0340	0.0521	0.1043
	0.8611	0.8882	1.0302	1.0759	1.8373
	0.7740	0.7303	0.8851	0.8535	1.3447
WFG9	0.8625	0.8886	1.0626	1.1364	1.8147
	0.0252	0.0226	0.0275	0.0629	0.0967
	0.9345	0.9337	1.1353	1.2594	1.9832
	0.8081	0.8430	1.0028	0.9760	1.4814

Table F.6: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{STD}$

PMGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	1.2378	1.4677	1.8540	2.2241	3.5681
	0.0186	0.0120	0.0412	0.0457	0.0781
	1.3201	1.4949	2.0809	2.3720	3.7919
	1.2176	1.4333	1.7861	2.1232	3.3336
DTLZ2	1.2569	1.4956	1.8959	2.1943	3.3468
	0.0129	0.0207	0.0414	0.0711	0.1381
	1.2871	1.5487	1.9783	2.3427	3.6720
	1.2178	1.4561	1.8068	2.0229	3.0707
DTLZ3	1.1530	1.3874	1.7730	2.0190	3.2190
	0.0325	0.0308	0.0540	0.0867	0.2008
	1.2553	1.4573	1.9270	2.2667	3.6683
	1.1003	1.3266	1.6035	1.7895	2.6980
DTLZ4	1.2952	1.2374	1.5956	2.4288	3.8916
	0.0077	0.0273	0.0530	0.0706	0.1439
	1.3062	1.2887	1.6958	2.5923	4.1724
	1.2480	1.1606	1.4160	2.2664	3.5558
DTLZ5	1.2078	1.3661	1.6833	1.9390	2.8974
	0.0139	0.0275	0.0419	0.0644	0.0999
	1.2441	1.4382	1.7781	2.0801	3.0902
	1.1675	1.2678	1.5843	1.7728	2.5608
DTLZ6	0.7531	1.0583	1.2928	1.3191	1.8911
	0.0091	0.0130	0.0302	0.0533	0.1054
	0.7702	1.0900	1.3568	1.4394	2.2008
	0.7240	1.0285	1.2083	1.1916	1.6136

Table F.6: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{STD}$ (continue)

PMGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.9799	1.1850	1.2829	1.3751	1.7269
	0.0721	0.0927	0.1777	0.3175	0.5383
	1.1784	1.3671	1.6896	2.1026	2.8645
	0.8311	0.9121	0.8358	0.7338	0.7661
WFG1	0.2960	0.1268	0.0253	0.0038	0.0000
	0.0261	0.0177	0.0071	0.0014	0.0000
	0.3653	0.1598	0.0455	0.0093	0.0000
	0.2066	0.0720	0.0090	0.0012	0.0000
WFG2	1.0264	1.1179	1.3727	1.5838	2.9955
	0.0191	0.0386	0.0628	0.1027	0.1739
	1.0672	1.1980	1.5057	1.7717	3.3403
	0.9666	1.0414	1.2426	1.2986	2.6030
WFG3	0.8921	0.9476	1.1133	1.2051	2.1662
	0.0222	0.0317	0.0617	0.0925	0.1429
	0.9543	1.0263	1.2478	1.4536	2.4326
	0.8375	0.8743	0.9449	0.9827	1.8330
WFG4	0.6532	0.6099	0.6093	0.6183	1.3725
	0.0168	0.0199	0.0260	0.0348	0.0666
	0.6900	0.6501	0.6836	0.7044	1.4960
	0.6056	0.5642	0.5516	0.5506	1.2262
WFG5	0.6618	0.8022	0.8037	0.7898	1.1734
	0.0311	0.0312	0.0470	0.0658	0.0932
	0.7412	0.8827	0.9427	1.0194	1.4512
	0.5780	0.7197	0.7128	0.6413	0.8868

Table F.6: Average, Standard Deviation, Maximum, and Minimum HV for the PMGPSO$_{STD}$ (continue)

PMGPSO$_{STD}$ HV					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.7509	0.7744	0.8707	0.9228	1.4395
	0.0272	0.0323	0.0452	0.0569	0.0940
	0.8093	0.8412	0.9773	1.0573	1.6454
	0.7046	0.7017	0.7690	0.8047	1.1628
WFG7	0.7014	0.7403	0.8493	0.8902	1.4708
	0.0327	0.0236	0.0372	0.0554	0.0859
	0.7821	0.8072	0.9330	0.9969	1.6911
	0.6426	0.6925	0.7307	0.7748	1.2506
WFG8	0.8172	0.8189	0.9521	1.0189	1.5799
	0.0217	0.0316	0.0447	0.0710	0.1133
	0.8711	0.8893	1.0841	1.1943	1.8373
	0.7577	0.7303	0.8395	0.8535	1.3341
WFG9	0.8599	0.8883	1.0642	1.1382	1.8140
	0.0263	0.0235	0.0284	0.0591	0.0895
	0.9345	0.9377	1.1409	1.2594	1.9832
	0.8069	0.8430	1.0028	0.9760	1.4814

F.2 Inverted Generational Distance Values

The average, standard deviation, maximum, and minimum IGD performance measure values for each algorithm on each problem instance are listed in tables F.7 to F.12. Note that these performance measure values are associated with Chapter 6.

Table F.7: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_R$

KnMGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0012	0.0056	0.0098	0.0091	0.0096
	0.0002	0.0003	0.0007	0.0005	0.0006
	0.0017	0.0064	0.0111	0.0100	0.0106
	0.0010	0.0050	0.0082	0.0080	0.0084
DTLZ2	0.0186	0.0148	0.0152	0.0148	0.0198
	0.0003	0.0004	0.0003	0.0003	0.0003
	0.0192	0.0156	0.0158	0.0153	0.0206
	0.0180	0.0139	0.0143	0.0142	0.0192
DTLZ3	0.0114	0.0103	0.0145	0.0153	0.0211
	0.0005	0.0003	0.0004	0.0003	0.0006
	0.0124	0.0110	0.0151	0.0162	0.0224
	0.0104	0.0095	0.0139	0.0147	0.0201
DTLZ4	0.0214	0.0212	0.0288	0.0259	0.0265
	0.0022	0.0018	0.0003	0.0002	0.0002
	0.0238	0.0257	0.0299	0.0262	0.0269
	0.0189	0.0187	0.0285	0.0253	0.0262

Table F.7: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_R$ (continue)

KnMGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ5	0.0210	0.0162	0.0138	0.0136	0.0126
	0.0003	0.0004	0.0005	0.0005	0.0005
	0.0214	0.0169	0.0145	0.0144	0.0137
	0.0203	0.0154	0.0128	0.0126	0.0116
DTLZ6	0.0018	0.0040	0.0053	0.0066	0.0078
	0.0002	0.0006	0.0007	0.0006	0.0010
	0.0023	0.0053	0.0066	0.0081	0.0096
	0.0014	0.0031	0.0040	0.0053	0.0061
DTLZ7	0.1173	0.1373	0.2237	0.0926	0.1380
	0.0011	0.0015	0.0007	0.0003	0.0003
	0.1193	0.1407	0.2253	0.0931	0.1388
	0.1134	0.1330	0.2219	0.0922	0.1373
WFG1	0.0363	0.0474	0.0749	0.0757	0.1316
	0.0010	0.0009	0.0008	0.0010	0.0006
	0.0394	0.0497	0.0767	0.0785	0.1337
	0.0346	0.0461	0.0734	0.0742	0.1312
WFG2	0.0479	0.0594	0.0885	0.0844	0.1344
	0.0012	0.0005	0.0003	0.0003	0.0004
	0.0517	0.0609	0.0891	0.0850	0.1350
	0.0454	0.0589	0.0880	0.0836	0.1336
WFG3	0.0756	0.1412	0.2432	0.3117	0.4838
	0.0001	0.0001	0.0001	0.0001	0.0001
	0.0759	0.1413	0.2433	0.3119	0.4840
	0.0754	0.1410	0.2430	0.3115	0.4836

Table F.7: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_R$ (continue)

KnMGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG4	0.0882	0.1532	0.2711	0.3012	0.4678
	0.0001	0.0002	0.0007	0.0011	0.0014
	0.0883	0.1536	0.2727	0.3033	0.4701
	0.0881	0.1529	0.2697	0.2975	0.4652
WFG5	0.0885	0.1542	0.2730	0.3029	0.4669
	0.0004	0.0001	0.0001	0.0002	0.0003
	0.0893	0.1545	0.2734	0.3033	0.4679
	0.0876	0.1539	0.2728	0.3026	0.4664
WFG6	0.0915	0.1556	0.2726	0.3012	0.4650
	0.0002	0.0002	0.0002	0.0003	0.0006
	0.0918	0.1560	0.2729	0.3020	0.4661
	0.0911	0.1553	0.2720	0.3007	0.4635
WFG7	0.0902	0.1551	0.2734	0.3019	0.4654
	0.0002	0.0002	0.0003	0.0003	0.0006
	0.0905	0.1554	0.2738	0.3024	0.4669
	0.0898	0.1547	0.2724	0.3012	0.4644
WFG8	0.0922	0.1564	0.2729	0.3016	0.4650
	0.0001	0.0002	0.0003	0.0003	0.0006
	0.0925	0.1568	0.2735	0.3024	0.4661
	0.0919	0.1560	0.2725	0.3008	0.4637
WFG9	0.0910	0.1548	0.2730	0.3024	0.4661
	0.0002	0.0002	0.0001	0.0002	0.0003
	0.0912	0.1554	0.2733	0.3028	0.4669
	0.0904	0.1544	0.2727	0.3020	0.4653

Table F.8: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{RI}$

KnMGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0012	0.0055	0.0098	0.0090	0.0096
	0.0002	0.0003	0.0006	0.0006	0.0006
	0.0018	0.0064	0.0111	0.0100	0.0106
	0.0010	0.0050	0.0082	0.0069	0.0084
DTLZ2	0.0188	0.0149	0.0152	0.0148	0.0198
	0.0004	0.0004	0.0003	0.0002	0.0003
	0.0194	0.0156	0.0159	0.0153	0.0206
	0.0180	0.0139	0.0143	0.0142	0.0192
DTLZ3	0.0114	0.0102	0.0145	0.0154	0.0211
	0.0005	0.0003	0.0004	0.0004	0.0006
	0.0127	0.0110	0.0153	0.0165	0.0224
	0.0102	0.0095	0.0137	0.0147	0.0197
DTLZ4	0.0215	0.0210	0.0288	0.0258	0.0265
	0.0022	0.0018	0.0002	0.0002	0.0001
	0.0238	0.0257	0.0299	0.0262	0.0269
	0.0184	0.0182	0.0285	0.0252	0.0262
DTLZ5	0.0210	0.0162	0.0139	0.0136	0.0127
	0.0003	0.0005	0.0005	0.0004	0.0005
	0.0214	0.0172	0.0148	0.0144	0.0137
	0.0203	0.0148	0.0128	0.0126	0.0116
DTLZ6	0.0018	0.0039	0.0053	0.0066	0.0076
	0.0002	0.0005	0.0006	0.0007	0.0009
	0.0023	0.0053	0.0066	0.0082	0.0096
	0.0014	0.0031	0.0040	0.0051	0.0058

Table F.8: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{RI}$ (continue)

KnMGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1170	0.1374	0.2236	0.0926	0.1380
	0.0015	0.0017	0.0007	0.0003	0.0003
	0.1197	0.1420	0.2253	0.0934	0.1388
	0.1121	0.1330	0.2219	0.0921	0.1373
WFG1	0.0364	0.0475	0.0750	0.0755	0.1315
	0.0011	0.0008	0.0008	0.0009	0.0007
	0.0394	0.0498	0.0773	0.0785	0.1343
	0.0343	0.0461	0.0734	0.0739	0.1285
WFG2	0.0478	0.0594	0.0884	0.0844	0.1344
	0.0009	0.0005	0.0004	0.0004	0.0004
	0.0517	0.0609	0.0891	0.0862	0.1350
	0.0454	0.0588	0.0874	0.0834	0.1335
WFG3	0.0756	0.1411	0.2431	0.3117	0.4838
	0.0001	0.0001	0.0001	0.0001	0.0001
	0.0759	0.1413	0.2433	0.3119	0.4841
	0.0753	0.1410	0.2430	0.3114	0.4836
WFG4	0.0882	0.1533	0.2711	0.3013	0.4678
	0.0001	0.0002	0.0007	0.0009	0.0014
	0.0883	0.1538	0.2727	0.3033	0.4707
	0.0880	0.1529	0.2697	0.2975	0.4646
WFG5	0.0885	0.1542	0.2730	0.3029	0.4669
	0.0004	0.0002	0.0002	0.0002	0.0003
	0.0893	0.1547	0.2734	0.3033	0.4679
	0.0876	0.1537	0.2728	0.3025	0.4663

Table F.8: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{RI}$ (continue)

KnMGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0915	0.1556	0.2725	0.3012	0.4650
	0.0002	0.0002	0.0002	0.0003	0.0005
	0.0918	0.1562	0.2729	0.3020	0.4661
	0.0911	0.1548	0.2720	0.3006	0.4635
WFG7	0.0902	0.1551	0.2733	0.3018	0.4654
	0.0002	0.0002	0.0003	0.0003	0.0006
	0.0905	0.1554	0.2738	0.3024	0.4669
	0.0898	0.1547	0.2724	0.3009	0.4643
WFG8	0.0921	0.1564	0.2729	0.3016	0.4650
	0.0002	0.0002	0.0003	0.0003	0.0006
	0.0925	0.1569	0.2735	0.3025	0.4665
	0.0918	0.1559	0.2723	0.3008	0.4637
WFG9	0.0910	0.1549	0.2730	0.3024	0.4661
	0.0002	0.0002	0.0001	0.0002	0.0003
	0.0912	0.1555	0.2733	0.3028	0.4669
	0.0904	0.1544	0.2727	0.3020	0.4653

Table F.9: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{STD}$

KnMGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0020	0.0055	0.0097	0.0090	0.0096
	0.0017	0.0003	0.0007	0.0007	0.0007
	0.0087	0.0064	0.0111	0.0106	0.0107
	0.0010	0.0048	0.0067	0.0062	0.0068
DTLZ2	0.0188	0.0148	0.0151	0.0148	0.0198
	0.0004	0.0004	0.0003	0.0002	0.0003
	0.0194	0.0156	0.0159	0.0154	0.0206
	0.0179	0.0139	0.0143	0.0142	0.0192
DTLZ3	0.0125	0.0103	0.0146	0.0154	0.0211
	0.0024	0.0004	0.0004	0.0004	0.0006
	0.0230	0.0114	0.0158	0.0166	0.0224
	0.0101	0.0095	0.0137	0.0147	0.0197
DTLZ4	0.0215	0.0212	0.0287	0.0258	0.0265
	0.0022	0.0017	0.0004	0.0002	0.0002
	0.0238	0.0257	0.0299	0.0262	0.0269
	0.0184	0.0182	0.0267	0.0252	0.0261
DTLZ5	0.0209	0.0162	0.0139	0.0136	0.0126
	0.0003	0.0005	0.0005	0.0004	0.0006
	0.0214	0.0172	0.0152	0.0144	0.0140
	0.0199	0.0148	0.0128	0.0126	0.0111
DTLZ6	0.0018	0.0039	0.0053	0.0066	0.0076
	0.0002	0.0005	0.0006	0.0008	0.0010
	0.0023	0.0053	0.0066	0.0082	0.0096
	0.0013	0.0031	0.0040	0.0051	0.0057

Table F.9: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{STD}$ (continue)

KnMGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.1165	0.1385	0.2240	0.0928	0.1382
	0.0016	0.0025	0.0011	0.0004	0.0005
	0.1197	0.1467	0.2267	0.0941	0.1403
	0.1120	0.1330	0.2219	0.0921	0.1373
WFG1	0.0355	0.0473	0.0752	0.0760	0.1316
	0.0015	0.0008	0.0011	0.0012	0.0008
	0.0394	0.0498	0.0792	0.0789	0.1343
	0.0329	0.0457	0.0731	0.0739	0.1285
WFG2	0.0477	0.0593	0.0883	0.0843	0.1344
	0.0009	0.0005	0.0004	0.0005	0.0004
	0.0517	0.0609	0.0891	0.0862	0.1352
	0.0454	0.0580	0.0872	0.0834	0.1331
WFG3	0.0756	0.1411	0.2432	0.3117	0.4838
	0.0001	0.0001	0.0001	0.0001	0.0001
	0.0759	0.1413	0.2433	0.3119	0.4841
	0.0753	0.1409	0.2430	0.3114	0.4835
WFG4	0.0882	0.1533	0.2711	0.3012	0.4676
	0.0001	0.0002	0.0007	0.0009	0.0014
	0.0883	0.1539	0.2727	0.3033	0.4707
	0.0880	0.1528	0.2693	0.2975	0.4636
WFG5	0.0884	0.1541	0.2730	0.3029	0.4669
	0.0005	0.0002	0.0001	0.0002	0.0003
	0.0899	0.1547	0.2734	0.3033	0.4679
	0.0873	0.1537	0.2727	0.3025	0.4663

Table F.9: Average, Standard Deviation, Maximum, and Minimum IGD for the KnMGPSO$_{STD}$ (continue)

KnMGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0914	0.1555	0.2725	0.3012	0.4650
	0.0003	0.0002	0.0002	0.0003	0.0005
	0.0918	0.1562	0.2729	0.3020	0.4661
	0.0907	0.1548	0.2720	0.3006	0.4635
WFG7	0.0902	0.1551	0.2733	0.3018	0.4654
	0.0002	0.0002	0.0003	0.0004	0.0006
	0.0905	0.1554	0.2738	0.3024	0.4676
	0.0897	0.1546	0.2724	0.3007	0.4643
WFG8	0.0921	0.1564	0.2729	0.3016	0.4650
	0.0002	0.0002	0.0003	0.0003	0.0006
	0.0925	0.1569	0.2739	0.3026	0.4665
	0.0918	0.1559	0.2723	0.3008	0.4637
WFG9	0.0910	0.1549	0.2729	0.3024	0.4661
	0.0002	0.0002	0.0001	0.0002	0.0003
	0.0912	0.1555	0.2733	0.3028	0.4669
	0.0904	0.1544	0.2725	0.3020	0.4653

Table F.10: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_R$

PMGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0033	0.0079	0.0125	0.0115	0.0122
	0.0002	0.0004	0.0011	0.0009	0.0014
	0.0039	0.0087	0.0140	0.0128	0.0150
	0.0030	0.0070	0.0101	0.0099	0.0094
DTLZ2	0.0047	0.0081	0.0135	0.0145	0.0209
	0.0002	0.0002	0.0003	0.0003	0.0003
	0.0051	0.0084	0.0141	0.0150	0.0215
	0.0045	0.0077	0.0129	0.0141	0.0201
DTLZ3	0.0043	0.0077	0.0136	0.0151	0.0212
	0.0002	0.0002	0.0003	0.0003	0.0003
	0.0048	0.0080	0.0142	0.0158	0.0217
	0.0038	0.0073	0.0130	0.0145	0.0207
DTLZ4	0.0102	0.0216	0.0287	0.0254	0.0263
	0.0006	0.0013	0.0002	0.0002	0.0002
	0.0116	0.0240	0.0290	0.0258	0.0265
	0.0092	0.0195	0.0277	0.0246	0.0258
DTLZ5	0.0048	0.0059	0.0065	0.0074	0.0075
	0.0005	0.0006	0.0005	0.0006	0.0007
	0.0064	0.0070	0.0073	0.0088	0.0098
	0.0040	0.0044	0.0053	0.0063	0.0065
DTLZ6	0.0015	0.0039	0.0053	0.0065	0.0070
	0.0001	0.0005	0.0006	0.0006	0.0006
	0.0019	0.0048	0.0066	0.0081	0.0080
	0.0013	0.0030	0.0043	0.0051	0.0057

Table F.10: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_R$ (continue)

PMGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.0968	0.1305	0.2210	0.0921	0.1375
	0.0003	0.0003	0.0003	0.0001	0.0001
	0.0978	0.1309	0.2215	0.0924	0.1378
	0.0962	0.1300	0.2203	0.0919	0.1372
WFG1	0.0331	0.0472	0.0767	0.0785	0.1348
	0.0004	0.0003	0.0003	0.0004	0.0003
	0.0344	0.0479	0.0773	0.0792	0.1354
	0.0326	0.0466	0.0760	0.0776	0.1343
WFG2	0.0392	0.0531	0.0825	0.0802	0.1300
	0.0004	0.0003	0.0003	0.0004	0.0003
	0.0401	0.0543	0.0832	0.0810	0.1306
	0.0381	0.0526	0.0817	0.0793	0.1293
WFG3	0.0721	0.1394	0.2425	0.3114	0.4836
	0.0003	0.0001	0.0001	0.0001	0.0001
	0.0729	0.1397	0.2427	0.3117	0.4837
	0.0716	0.1391	0.2423	0.3112	0.4834
WFG4	0.0874	0.1532	0.2714	0.3010	0.4642
	0.0002	0.0003	0.0005	0.0009	0.0010
	0.0877	0.1537	0.2731	0.3027	0.4659
	0.0871	0.1526	0.2706	0.2995	0.4622
WFG5	0.0872	0.1540	0.2730	0.3026	0.4664
	0.0002	0.0001	0.0002	0.0002	0.0003
	0.0876	0.1543	0.2734	0.3032	0.4670
	0.0869	0.1537	0.2727	0.3022	0.4659

Table F.10: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_R$ (continue)

PMGPSO$_R$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0873	0.1538	0.2721	0.3010	0.4649
	0.0002	0.0002	0.0002	0.0002	0.0005
	0.0878	0.1542	0.2725	0.3014	0.4658
	0.0869	0.1535	0.2718	0.3006	0.4636
WFG7	0.0877	0.1541	0.2728	0.3016	0.4652
	0.0002	0.0002	0.0002	0.0003	0.0005
	0.0881	0.1544	0.2734	0.3021	0.4663
	0.0873	0.1536	0.2725	0.3008	0.4640
WFG8	0.0877	0.1539	0.2720	0.3011	0.4647
	0.0004	0.0004	0.0003	0.0003	0.0004
	0.0888	0.1548	0.2725	0.3016	0.4655
	0.0870	0.1533	0.2714	0.3006	0.4636
WFG9	0.0868	0.1532	0.2724	0.3020	0.4657
	0.0002	0.0001	0.0001	0.0001	0.0002
	0.0871	0.1535	0.2726	0.3023	0.4661
	0.0864	0.1530	0.2722	0.3017	0.4653

Table F.11: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{RI}$

PMGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0033	0.0078	0.0127	0.0115	0.0122
	0.0002	0.0004	0.0009	0.0008	0.0012
	0.0039	0.0087	0.0143	0.0131	0.0150
	0.0029	0.0070	0.0101	0.0093	0.0094
DTLZ2	0.0047	0.0081	0.0135	0.0146	0.0208
	0.0002	0.0002	0.0003	0.0003	0.0004
	0.0051	0.0087	0.0145	0.0151	0.0217
	0.0044	0.0077	0.0127	0.0141	0.0198
DTLZ3	0.0043	0.0077	0.0136	0.0150	0.0212
	0.0002	0.0002	0.0003	0.0003	0.0003
	0.0048	0.0081	0.0145	0.0158	0.0220
	0.0038	0.0073	0.0130	0.0144	0.0203
DTLZ4	0.0103	0.0215	0.0287	0.0254	0.0263
	0.0006	0.0013	0.0003	0.0003	0.0002
	0.0117	0.0244	0.0292	0.0259	0.0266
	0.0092	0.0188	0.0276	0.0244	0.0258
DTLZ5	0.0048	0.0060	0.0064	0.0074	0.0075
	0.0005	0.0006	0.0005	0.0005	0.0007
	0.0064	0.0071	0.0073	0.0088	0.0098
	0.0039	0.0044	0.0050	0.0063	0.0061
DTLZ6	0.0015	0.0038	0.0052	0.0065	0.0070
	0.0001	0.0004	0.0006	0.0006	0.0007
	0.0019	0.0048	0.0066	0.0081	0.0082
	0.0012	0.0030	0.0042	0.0049	0.0052

Table F.11: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{RI}$ (continue)

PMGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.0968	0.1304	0.2209	0.0921	0.1374
	0.0004	0.0003	0.0003	0.0001	0.0002
	0.0985	0.1311	0.2215	0.0924	0.1378
	0.0959	0.1296	0.2200	0.0918	0.1370
WFG1	0.0331	0.0473	0.0767	0.0785	0.1347
	0.0003	0.0004	0.0004	0.0004	0.0003
	0.0344	0.0482	0.0775	0.0792	0.1354
	0.0326	0.0466	0.0760	0.0776	0.1341
WFG2	0.0392	0.0531	0.0825	0.0802	0.1300
	0.0004	0.0004	0.0003	0.0004	0.0003
	0.0403	0.0543	0.0832	0.0810	0.1309
	0.0381	0.0522	0.0817	0.0793	0.1293
WFG3	0.0720	0.1394	0.2425	0.3114	0.4836
	0.0003	0.0001	0.0001	0.0001	0.0001
	0.0729	0.1397	0.2427	0.3117	0.4839
	0.0716	0.1391	0.2423	0.3112	0.4834
WFG4	0.0873	0.1532	0.2715	0.3011	0.4641
	0.0001	0.0003	0.0005	0.0007	0.0010
	0.0877	0.1537	0.2731	0.3027	0.4659
	0.0870	0.1526	0.2705	0.2995	0.4620
WFG5	0.0872	0.1540	0.2730	0.3027	0.4664
	0.0002	0.0001	0.0002	0.0002	0.0003
	0.0876	0.1544	0.2734	0.3032	0.4670
	0.0868	0.1537	0.2727	0.3022	0.4659

Table F.11: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{RI}$ (continue)

PMGPSO$_{RI}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0873	0.1538	0.2721	0.3010	0.4649
	0.0003	0.0002	0.0002	0.0002	0.0004
	0.0878	0.1542	0.2725	0.3015	0.4658
	0.0868	0.1535	0.2718	0.3004	0.4636
WFG7	0.0877	0.1541	0.2728	0.3016	0.4652
	0.0002	0.0002	0.0002	0.0003	0.0005
	0.0883	0.1544	0.2734	0.3021	0.4663
	0.0873	0.1536	0.2723	0.3008	0.4640
WFG8	0.0877	0.1538	0.2720	0.3011	0.4647
	0.0004	0.0004	0.0003	0.0002	0.0004
	0.0888	0.1548	0.2726	0.3016	0.4655
	0.0869	0.1532	0.2714	0.3006	0.4636
WFG9	0.0867	0.1532	0.2724	0.3020	0.4657
	0.0002	0.0001	0.0001	0.0001	0.0002
	0.0871	0.1535	0.2726	0.3023	0.4663
	0.0863	0.1528	0.2721	0.3017	0.4650

Table F.12: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{STD}$

PMGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ1	0.0033	0.0078	0.0126	0.0113	0.0120
	0.0002	0.0004	0.0012	0.0011	0.0013
	0.0039	0.0087	0.0143	0.0131	0.0150
	0.0028	0.0070	0.0076	0.0062	0.0090
DTLZ2	0.0047	0.0081	0.0135	0.0146	0.0208
	0.0002	0.0002	0.0003	0.0003	0.0004
	0.0052	0.0087	0.0145	0.0155	0.0217
	0.0044	0.0077	0.0127	0.0141	0.0198
DTLZ3	0.0043	0.0077	0.0136	0.0150	0.0212
	0.0002	0.0002	0.0003	0.0003	0.0003
	0.0048	0.0084	0.0145	0.0158	0.0220
	0.0038	0.0073	0.0130	0.0144	0.0203
DTLZ4	0.0104	0.0215	0.0287	0.0254	0.0263
	0.0006	0.0013	0.0003	0.0003	0.0002
	0.0121	0.0244	0.0292	0.0259	0.0266
	0.0092	0.0183	0.0276	0.0244	0.0258
DTLZ5	0.0047	0.0060	0.0064	0.0073	0.0074
	0.0005	0.0006	0.0005	0.0006	0.0007
	0.0064	0.0074	0.0076	0.0088	0.0098
	0.0039	0.0044	0.0048	0.0060	0.0061
DTLZ6	0.0015	0.0038	0.0052	0.0064	0.0070
	0.0002	0.0004	0.0006	0.0006	0.0008
	0.0019	0.0048	0.0066	0.0081	0.0091
	0.0010	0.0030	0.0040	0.0049	0.0051

Table F.12: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{STD}$ (continue)

PMGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
DTLZ7	0.0968	0.1305	0.2209	0.0921	0.1374
	0.0004	0.0003	0.0004	0.0001	0.0002
	0.0985	0.1316	0.2217	0.0925	0.1378
	0.0959	0.1296	0.2200	0.0918	0.1369
WFG1	0.0331	0.0474	0.0769	0.0787	0.1347
	0.0004	0.0005	0.0006	0.0006	0.0003
	0.0349	0.0491	0.0788	0.0804	0.1354
	0.0324	0.0463	0.0750	0.0771	0.1338
WFG2	0.0392	0.0531	0.0825	0.0802	0.1300
	0.0005	0.0003	0.0003	0.0004	0.0003
	0.0403	0.0543	0.0833	0.0810	0.1309
	0.0381	0.0522	0.0817	0.0793	0.1293
WFG3	0.0720	0.1394	0.2425	0.3114	0.4836
	0.0003	0.0001	0.0001	0.0001	0.0001
	0.0729	0.1398	0.2428	0.3117	0.4839
	0.0716	0.1391	0.2423	0.3112	0.4834
WFG4	0.0874	0.1532	0.2714	0.3010	0.4641
	0.0001	0.0002	0.0006	0.0007	0.0010
	0.0877	0.1537	0.2731	0.3027	0.4664
	0.0870	0.1526	0.2699	0.2995	0.4620
WFG5	0.0871	0.1540	0.2730	0.3027	0.4663
	0.0002	0.0001	0.0002	0.0002	0.0003
	0.0876	0.1544	0.2734	0.3032	0.4670
	0.0866	0.1537	0.2725	0.3022	0.4657

Table F.12: Average, Standard Deviation, Maximum, and Minimum IGD for the PMGPSO$_{STD}$ (continue)

PMGPSO$_{STD}$ IGD					
Benchmark Function	n_m				
	3	**5**	**8**	**10**	**15**
WFG6	0.0873	0.1538	0.2721	0.3010	0.4649
	0.0003	0.0002	0.0002	0.0002	0.0004
	0.0878	0.1542	0.2725	0.3016	0.4658
	0.0868	0.1534	0.2717	0.3004	0.4636
WFG7	0.0878	0.1541	0.2728	0.3016	0.4652
	0.0003	0.0002	0.0002	0.0003	0.0005
	0.0885	0.1545	0.2734	0.3021	0.4663
	0.0873	0.1536	0.2723	0.3008	0.4640
WFG8	0.0878	0.1538	0.2720	0.3012	0.4647
	0.0005	0.0004	0.0003	0.0003	0.0004
	0.0893	0.1548	0.2726	0.3018	0.4655
	0.0869	0.1529	0.2714	0.3005	0.4636
WFG9	0.0867	0.1532	0.2724	0.3020	0.4657
	0.0002	0.0001	0.0001	0.0001	0.0002
	0.0871	0.1535	0.2726	0.3023	0.4663
	0.0863	0.1528	0.2721	0.3017	0.4650

Appendix G

Derived Publications

The following publication was derived from this dissertation.

Cian Steenkamp, Andries P. Engelbrecht, A Scalability Study of the Multi-guide Particle Swarm Optimization Algorithm. Submitted to the Swarm and Evolutionary Computation Journal, 2021.